8-2011

Pseudocodewords of Parity-Check Codes

Wittawat Kositwattanarerk
Clemson University, wkositw@clemson.edu

Pseudocodewords of Parity-Check Codes

A Dissertation
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
Mathematical Sciences

---

by
Wittawat Kositwattanarerk
August 2011

Accepted by:
Dr. Gretchen L. Matthews, Committee Chair
Dr. Shuhong Gao
Dr. Neil J. Calkin
Dr. Elena S. Dimitrova

Abstract

The success of modern algorithms for the decoding problem such as message-passing iterative decoding and linear programming decoding lies in their local nature. This feature allows the algorithms to be extremely fast and capable of correcting more errors than guaranteed by the classical minimum distance of the code. Nonetheless, the performance of these decoders depends crucially on the Tanner graph representation of the code. In order to understand this choice of representation, we need to analyze the pseudocodewords of the Tanner graph of a code. These pseudocodewords are outputs of local decoding algorithms which may not be legitimate codewords. In this dissertation, we introduce a lifted fundamental cone and show that there is a one-to-one correspondence between graph cover pseudocodewords of a binary code and integer points in the lifted fundamental cone. We use this fact to prove the rationality of the generating function of the pseudocodewords for a general binary parity-check code. Our approach also yields algorithms for producing this generating function and provides tools for studying the irreducible pseudocodewords. Understanding irreducible pseudocodewords is crucial to determining the best representation of a code. Moreover, combining these techniques with the recent characterization of fundamental cone over $\mathbb{F}_3$, we can analyze ternary parity-check codes. Finally, we make progress in the study of more general nonbinary codes by determining constraints satisfied by all pseudocodewords of a code over $\mathbb{F}_p$ where p is prime.

Acknowledgments

Thank you for your interest in this dissertation. I hope your journey through this manuscript is an enlightening one.

This dissertation would not have been possible without the supervision of my advisor, Gretchen L. Matthews. Throughout my years at Clemson University, Dr. Matthews has not only taught me invaluable research skills, she takes genuine interest in my professional development and provides the necessary directions as I advance in my career. I am very grateful for her wisdom, support, and guidance.

It is an honor to have Shuhong Gao, Neil J. Calkin, and Elena S. Dimitrova on my dissertation committee. I wish to express my gratitude for their helpful insights.

Many individuals contribute to the results in this dissertation. I would like to thank my and Dr. Matthews' colleagues, especially Judy L. Walker, Christine A. Kelley, Pascal O. Vontobel, Vitaly Skachek, Wen-Ching Winnie Li, Raymond Hemmecke, and Matthias Köppe for their suggestions and comments.

I am grateful to Thann N. Ward, my undergraduate advisor at the University of Virginia. His passion and his vast knowledge in algebraic coding theory ignited my interest in this area.

I consider my mathematical career to have begun at a very early age. I am grateful for all of my teachers, mentors, and my sponsor. I would not be able to live up to my fullest potential if not for their investment in the little boy I was.

I would also like to thank my friends for making my life as a graduate student fun. I value highly my colleagues at Clemson for their companionship, inspiration, as well as occasional distraction.

My deepest thanks, however, go to my family. I am indebted to my parents for their unconditional love and to my brother and sisters for their constant support.

Table of Contents

List of Figures

Chapter 1

Introduction

1.1 Historical Background

Coding theory is the study of how information can be transmitted efficiently and reliably. The usual practice involves encoding data as a string of code symbols where the structure of the code allows detection and correction of errors. One of the major milestones in coding theory is the invention of iterative decoders and the linear programming decoder. For block codes, these decoders search for a word that satisfies each parity condition iteratively rather than a word that satisfies every parity condition collectively. The result is a class of fast and efficient algorithms for the decoding problem. In addition, these algorithms are capable of correcting more errors than guaranteed by the classical minimum distance of the code. When performed on low-density parity-check (LDPC) codes, iterative decoding allow transmission of information at rates up to the Shannon limit under several channels. Nonetheless, the major drawback of these algorithms is that they may yield a noncodeword output called a pseudocodeword. The focus of this dissertation is the analysis, characterization, and enumeration of these pseudocodewords.

The prototype of iterative decoders was invented by Gallager [15] in 1962. Unfortunately, the complexity of the proposed algorithms exceeded the capabilities of the computers at that time. In 1981, Tanner [38] pursued the practical implementation of the work of Gallager; however, their contributions remained dormant for another decade. It was not until turbo codes were introduced by Berrou, Glavieux, and Thitimajshima [8] in 1993 that practical implementation of iterative decoders surfaced. In particular, iterative decoders received the attention they deserved when Mackay and Neal [31], and Richardson, Shokrollahi, and Urbanke [32] demonstrated that these decoding algorithms enable communication at rates near the channel capacity under several circumstances.

Among the very first to study the behavior of iterative decoders and their noncodeword outputs is Wiberg [41] in his 1996 dissertation. Many attempts followed soon after [1, 11, 25, 27, 28, 29]. In [28], Koetter and Vontobel introduced an object called the fundamental cone which contains all graph cover pseudocodewords. Linear programming decoding introduced by Feldman, Wainwright, and Karger [13] suggested a similar explanation for the noncodeword outputs. In [27], Koetter, Li, Vontobel, and Walker characterized all the pseudocodewords within the fundamental cone; in addition, they proved that the pseudocodewords of a cycle code correspond to the monomials appearing (with nonzero coefficient) in an expansion of a rational function, specifically the edge zeta function of the normal graph of the code. Prompted by these results for cycle codes, the quest to determine such a rational function for a general parity-check code is the heart of this dissertation as well as the focus of other research [30].

1.2 Contributions of this Dissertation

In this dissertation, methods from discrete geometry are exploited to give a rational generating function for the pseudocodewords of a binary parity-check code and to provide tools to study pseudocodewords. We introduce a lifted fundamental cone $\hat{\mathcal{K}}$; the fundamental cone mentioned earlier is a projection of $\hat{\mathcal{K}}$. The lifted cone has the advantage that its integer points are precisely the pseudocodewords. This allows us to prove that the generating function of the pseudocodewords of a general parity-check code is rational, a fact proved independently by Li, Lu, and Wang [30] via other methods. Our approach differs from that of [30] in that we use the lifted fundamental cone and appeal to monomial substitution methods while in [30] they rely on generators of the fundamental cone with even entries and inclusion-exclusion. As a result, we obtain simpler rational functions and the methods presented here yield algorithms for producing this generating function. In particular, Barvinok's algorithm, a breakthrough polynomial-time algorithm to count lattice points in a rational polytope of a given dimension [2], is utilized here. Because Barvinok's algorithm (and subsequent improvements) have been implemented in software such as Barvinok 0.27 [39], LattE [20], and LattE macchiato [22], this perspective gives rise to computational tools to study pseudocodewords. In addition, the lifted fundamental cone provides a framework for studying the irreducible pseudocodewords.

Because some applications require codes over larger alphabets, there is an interest in nonbinary parity-check codes. In this dissertation, we show that the results mentioned above also apply to ternary parity-check codes. In addition, we make progress on more general nonbinary codes by considering linear codes over $\mathbb{F}_p$ where p is prime.

1.3 Outline of the Dissertation

This chapter concludes with a summary of notation that will be used throughout the dissertation. Chapter 2 and Chapter 3 present basic results from coding theory and discrete geometry respectively. Chapter 2 outlines linear programming decoding, message-passing iterative decoding, graph cover decoding, and different types of pseudocodewords. Chapter 3 discusses the generating function of integer points in a rational cone and Barvinok's algorithm.

Chapters 4 and 5 deliver the main results of this dissertation. The definition of the lifted fundamental cone is presented in Chapter 4 along with the discussion of the generating function for the pseudocodewords. Chapter 5 investigates the pseudocodewords of a nonbinary code.

1.4 Notation

The sets of integers, rational numbers, and real numbers are denoted $\mathbb{Z}$, $\mathbb{Q}$, and $\mathbb{R}$ respectively. The finite field with p elements is denoted $\mathbb{F}_p$, and $\mathbb{F}_p^* = \mathbb{F}_p \setminus \{0\}$. If p is prime, the elements of $\mathbb{F}_p$ are written as $0, 1, \ldots, p-1$. In particular, the finite field with 2 elements is $\mathbb{F}_2 = \{0, 1\}$. If confusion is likely to arise, we use $\oplus$ and $\odot$ to denote finite field addition and multiplication.

The set of all $m \times n$ matrices with entries in A is denoted $A^{m \times n}$, and, as usual, $A^n := A^{1 \times n}$. Given a matrix $H \in A^{m \times n}$, H^T denotes the transpose of H, $Row_j(H)$ denotes the j^{th} row of H, and h_{ji} denotes the entry of H in the j^{th} row and i^{th} column. Vectors are displayed in bold and the i^{th} coordinate of a vector $\mathbf{v}$ is denoted v_i. Let $\mathbf{e}_i := (0, \ldots, 0, 1, 0, \ldots, 0)$ be the vector with 1 in the i^{th} coordinate and 0's elsewhere. The support of $\mathbf{v} \in A^n$ is $supp(\mathbf{v}) := \{i \mid v_i \neq 0\}$.

The graphs we consider in this dissertation are all simple graphs, meaning undirected graphs that have no loops and no more than one edge between any two different vertices. Given a vertex u of a graph G, the neighborhood of u, $Nbhd(u)$, is the set of vertices of G that are adjacent to u. The input size of $a \in \mathbb{Z}$, denoted by $\chi(a)$, is the number of bits needed to express a in binary. The standard convention is $\chi(a) = 1 + \log |a|$. However, one may wish to take into account the number of bits required to describe $\log |a|$; to do so, one may take $\chi(a) = 1 + \log |a| + O(\log \log |a|)$. We leave it to the reader to determine which definition of $\chi(a)$ is more appropriate for the given application. For this reason, all complexity results are given in terms of $\chi(a)$. Here and throughout the dissertation, the logarithm is taken base 2.

Chapter 2

Coding Theory Background

Since Shannon's landmark paper in 1948 [33], much progress has been made in the field of coding theory. In this chapter, we summarize some fundamental results which are relevant to the subject of this dissertation. Section 2.1 introduces basic definitions and terminologies from coding theory [19]. Sections 2.2, 2.3, and 2.4 outline linear programming decoding [13], message-passing iterative decoding [18, 21], and graph cover decoding [40] respectively. Pseudocodewords arising from these decoding algorithms are discussed in Section 2.5 [13, 27]. Finally, Section 2.6 introduces cycle codes and their properties [27].

2.1 Coding Theory

A *binary linear code* C of length n and dimension k is a subspace of $\mathbb{F}_2^n$ of dimension k. We often say code to mean binary linear code. Elements of C are called codewords. A *parity-check matrix* of a code C is a binary matrix $H \in \mathbb{F}_2^{r \times n}$ such that C is the null space of H. In other words, a parity-check matrix H of C has the

property that $\mathbf{y} \in \mathbb{F}_2^n$ is a codeword of C if and only if

$$H\mathbf{y}^T = \mathbf{0} \in \mathbb{F}_2^{r\times 1}.$$

We do not require that $r = n-k$; that is, H may not have full rank. As illustrated in Example 2.1 below, a parity-check matrix of a code is not unique. However, as we will see, some algorithms and related notions are sensitive to the choice of parity-check matrix. Thus, we use the notation $C(H)$ to emphasize that the code C is given by the parity-check matrix H.

Example 2.1 *Consider the matrices*

$$H_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}$$

and

$$H_2 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}.$$

Then,

$$C(H_1) = C(H_2)$$

but

$$H_1 \neq H_2.$$

One may note that

$$C(H_1) = C(H_2) = \{(0,0,0,0,0,0,0), (0,0,1,0,1,1,1), (0,1,0,1,0,1,1),$$
$$(0,1,1,1,1,0,0), (1,0,0,1,1,0,1), (1,0,1,1,0,1,0),$$
$$(1,1,0,0,1,1,0), (1,1,1,0,0,0,1)\}$$

is a binary linear code of length 7 *and dimension* 3. *Hence, while* $H_1 \neq H_2$, H_1 *and* H_2 *are parity-check matrices for the same code.*

We consider the problem of data communication over a memoryless binary input symmetric output channel. That is, the channel transmits binary data where the bit error probability is independent of the transmitting symbol and any previously transmitted symbols. Given a received word $\mathbf{w}$, the *maximum likelihood (ML) decoder* finds a codeword $\mathbf{y} \in C$ that maximizes the probability that $\mathbf{w}$ is received given that $\mathbf{y}$ is the transmitted codeword. More precisely, the ML decoder is the decision rule

$$\mathbf{c}_{ML} := \arg\max_{\mathbf{y} \in C} P(\mathbf{w}|\mathbf{y}).$$

The codeword $\mathbf{c}_{ML}$ is called the maximum likelihood codeword. Notice that the channel assumptions imply

$$\mathbf{c}_{ML} = \arg\max_{\mathbf{y} \in C} \prod_{i=1}^{n} P(w_i|y_i).$$

One may also define a *cost function* associated to a word $\mathbf{w} \in \mathbb{F}_2^n$ to be

$$\sum_{i=1}^{n} \gamma_i y_i \tag{2.1}$$

where

$$\gamma_i := \log\left(\frac{P(w_i|y_i=0)}{P(w_i|y_i=1)}\right)$$

denotes the log-likelihood ratio at the i^{th} coordinate. One may interpret γ_i as the "cost" of decoding $y_i = 1$. Therefore, the ML decoding problem can be rephrased as

$$\mathbf{c}_{ML} = \arg\min_{\mathbf{y}\in C} \sum_{i=1}^{n} \gamma_i y_i.$$

ML decoding is probably the most intuitive decoding scheme since it always outputs the codeword that in some sense best explains the received vector. Nonetheless, the ML decoding problem is known to be NP-hard for a general parity-check code [7]. In practice, a code C with large n, meaning a long code, needs to possess certain algebraic or geometric structure so that ML decoding is computationally feasible.

Practical approximations of ML decoding for a general parity-check code include linear programming decoding, message-passing iterative decoding, and graph cover decoding. Linear programming decoding performs ML decoding on $Q(H)$, an approximate of $C(H)$ which is more accessible via linear programming. Message-passing iterative decoding uses local cost functions in iterative low-complexity processes. Graph cover decoding is essentially equivalent to linear programming decoding and is a close approximation of message-passing iterative decoding. Except in a few special cases, these algorithms are not optimal; in particular, they may not output a codeword. These noncodeword outputs are called pseudocodewords and will be the main subject of this dissertation. In the following sections, we detail linear programming decoding, message-passing iterative decoding, and graph cover decoding.

2.2 Linear Programming Decoding

The problem of ML decoding for a code C of length n can be stated as a linear program. To do so, consider the code C as implicitly embedded in $\mathbb{R}^n$.

Definition 2.2 *Given a binary linear code C of length n, the codeword polytope of C is*

$$poly(C) := \left\{ \sum_{\mathbf{y} \in C} \lambda_{\mathbf{y}} \mathbf{y} \;\middle|\; \lambda_{\mathbf{y}} \geq 0, \sum_{\mathbf{y} \in C} \lambda_{\mathbf{y}} = 1 \right\} \subseteq [0,1]^n.$$

Notice that $poly(C)$ is the convex hull of the codewords of C. Moreover, the vertices of $poly(C)$ are precisely the codewords of C. Since the cost function (2.1) is linear, we may once again rephrase ML decoding as

$$\mathbf{c}_{ML} = \arg \min_{\mathbf{y} \in poly(C)} \sum_{i=1}^{n} \gamma_i y_i.$$

However, solving this problem as a linear program is still not practical for codes of reasonable length; the description of the constraints required to determine $poly(C)$ is typically exponential in block length.

In an effort to make this problem more computationally feasible, Feldman, Wainwright, and Karger [13] replace the codeword polytope with a relaxed polytope as described in Definition 2.3 below. While ML decoding of a code C does not depend on the choice of parity-check matrix for C, the choice of parity-check matrix does impact the relaxed polytope. Hence, we will see that the result is an approximation to ML decoding.

Definition 2.3 *Given $H \in \mathbb{F}_2^{r \times n}$, let $Q(H)$ be the intersection of the codeword polytopes of the r simple parity-check codes defined by the rows of H; that is,*

$$Q(H) := \cap_{j=1}^{r} poly(C(Row_j(H))).$$

Notice that the relaxed polytope depends not only on C but also the particular choice of parity-check matrix for C. It follows that

$$poly(C) \subseteq Q(H),$$

and $Q(H)$ has a more tractable representation than the original codeword polytope. The *linear programming (LP) decoder* is the decision rule

$$\mathbf{c}_{LP} := \arg \min_{\mathbf{y} \in Q(H)} \sum_{i=1}^{n} \gamma_i y_i.$$

The LP decoder has the ML certificate property: if the LP decoder outputs a codeword, then it is guaranteed to be the ML codeword. In other words, if $\mathbf{c}_{LP} \in C$, then $\mathbf{c}_{LP} = \mathbf{c}_{ML}$. Nonetheless, as the following example illustrates, it could be the case that

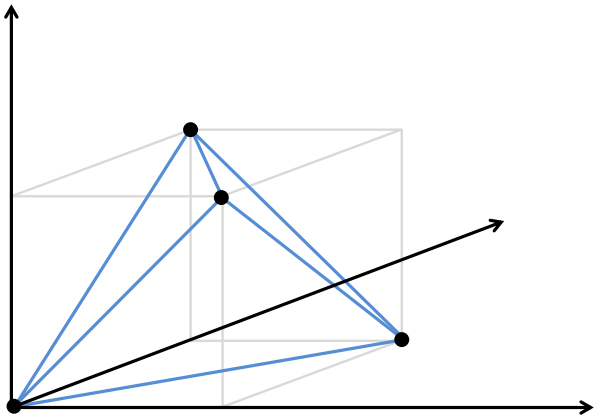
$$poly(C) \subset Q(H).$$

As a result, the LP decoder may yield a vertex of $Q(H)$ that is not in $poly(C)$. Hence, the LP decoder may yield an output that is not a codeword of $C = C(H)$.

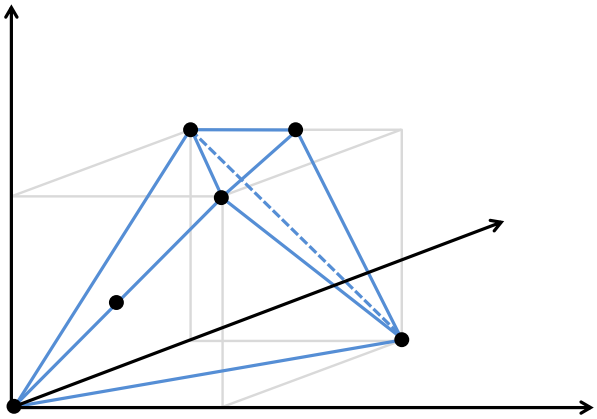
Example 2.4 *Consider the code $C(H)$ given by*

$$H = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix}.$$

The codewords of $C(H)$ are $(0,0,0,0)$, $(0,1,1,1)$, $(1,0,1,0)$, and $(1,1,0,1)$, and these vectors correspond precisely to the vertices of $poly(C)$. However, $Q(H)$ has 2 vertices, $\left(\frac{1}{2}, 1, \frac{1}{2}, 0\right)$ and $\left(\frac{1}{2}, 0, \frac{1}{2}, 1\right)$, that are not in $C(H)$ (and hence $poly(C)$). Three-dimensional projections of $poly(C)$ and $Q(H)$ are displayed in Figure 2.1.



$poly(C)$



$Q(H)$

Figure 2.1: Three-dimensional projections of $poly(C)$ and $Q(H)$ from Example 2.4

Definition 2.5 *Given $H \in \mathbb{F}_2^{r\times n}$, the LP pseudocodewords of the code $C(H)$ are the vertices of $Q(H)$.*

Notice that the LP pseudocodewords of $C(H)$ include the codewords of $C(H)$. The LP pseudocodewords will be discussed in more details in Section 2.5.

2.3 Message-Passing Iterative Decoding

In 1981, Tanner introduced a graphical representation of a parity-check matrix [38]. Intuitively, the columns of H are identified with the coordinates of words in $\mathbb{F}_2^n$ (more precisely, the coordinates of the codewords of $C(H)$) and the rows of H are identified with parity conditions, i.e. checks, that the codewords of $C(H)$ have to satisfy. This insight inspires a graphical representation of the parity-check matrix H. We make the construction precise as follows.

Definition 2.6 *Let $H \in \mathbb{F}_2^{r\times n}$. The Tanner graph of H, denoted $T(H)$, is a bipartite graph with vertex set $X \cup F$ such that*

- *Each vertex in $X = \{x_1, \ldots, x_n\}$ corresponds to a column of H and is called a bit node.*
- *Each vertex in $F = \{f_1, \ldots, f_r\}$ corresponds to a row of H and is called a check node.*
- *$\{x_i, f_j\}$ is an edge if and only if $h_{ji} = 1$.*

In other words, $T(H)$ is a bipartite graph with biadjacency matrix H, the vertex x_i corresponds to the i^{th} column of H, the vertex f_j corresponds to the j^{th} row of H, and the vertices x_i and f_j are adjacent if and only if $h_{ji} \neq 0$. We sometimes say the Tanner graph of $C(H)$ to mean $T(H)$.

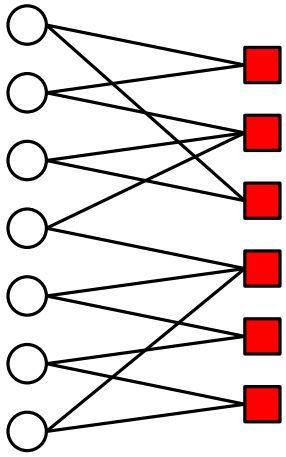


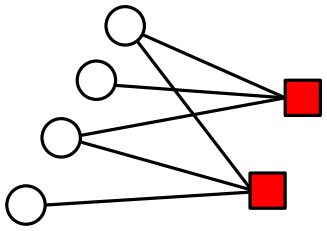
Figure 2.2: The Tanner graph $T(H)$ for the parity-check matrix H given in Example 2.7

Example 2.7 *Consider the code $C(H)$ given by*

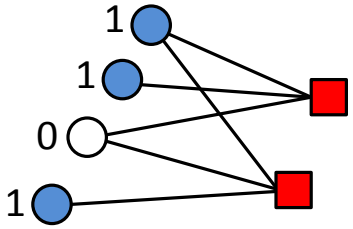
$$H = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}.$$

The Tanner graph of H is displayed in Figure 2.2 with the bit nodes represented by white circles and the check nodes represented by dark squares.

Notice that $\mathbf{c} = (c_1, c_2, \ldots, c_n)$ is a codeword of $C(H)$ if and only if the binary value assignment $(c_1, c_2, \ldots, c_n)$ to the bit nodes of the Tanner graph $T(H)$ makes the binary sum of the values at the neighbors of every check node zero. Thus, the



The Tanner graph of H



The codeword $(1, 1, 0, 1)$ on the Tanner graph of H

Figure 2.3: The Tanner graph of H and the codeword $(1, 1, 0, 1)$ from Example 2.8

Tanner graph $T(H)$ is a graphical model of the parity-check matrix H and hence the code $C(H)$.

Example 2.8 *Consider the code $C(H)$ from Example 2.4 given by*

$$H = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix}.$$

The Tanner graph of H and the codeword $(1, 1, 0, 1)$ of $C(H)$ are portrayed in Figure 2.3. The reader may verify that the binary sum of the values at the neighbors of every check node is zero.

A *message-passing iterative (MPI) decoder* for a code $C(H)$ takes as input the Tanner graph of H and the log-likelihood ratio $\boldsymbol{\gamma}$. Recall that the value γ_i can be interpreted as the "cost" of decoding node i to 1. For each iteration, a check node observes the log-likelihood ratio at its neighboring bit nodes and sends an updated ratio that satisfies the check node. The bit nodes update their costs and the process iterates. The min-sum algorithm can be described more precisely as follows.

- Initialization: For each bit node i, initialize the local cost γ_i. For each check node j and for all $s \in Nbhd(j)$, initialize $\mu_{j,s}^{(0)} := 0$.

- Iteration:

  For $i = 1, \ldots, m$:

  1. For all bit nodes s and for all $j \in Nbhd(s)$, bit-to-check messages are given by

  $$\mu_{s,j}^{(i)} := \gamma_s + \sum_{j' \in Nbhd(s) - \{j\}} \mu_{j',s}^{(i-1)}.$$

  2. For all check nodes j and for all $s \in Nbhd(j)$, check-to-bit messages are given by

  $$\mu_{j,s}^{(i)} := \prod_{s' \in Nbhd(j) - \{s\}} \operatorname{sgn}\left(\mu_{s',j}^{(i)}\right) \cdot \min_{s' \in Nbhd(j) - \{s\}} \left|\mu_{s',j}^{(i)}\right|$$

- Final cost computation: The final cost at the bit node i after m iterations is

$$\mu_i := \gamma_i + \sum_{j \in Nbhd(i)} \mu_{j,i}^{(m)}.$$

Finally, the message-passing iterative decoder makes the decision

$$\mathbf{c}_{MPI} := \arg\min_{\mathbf{y}\in\mathbb{F}_2^n} \sum_{i=1}^{n} \mu_i y_i.$$

The hallmark of this algorithm is that the entire process is *local*, meaning that the decision made at each vertex at any stage of the algorithm is based solely on the incoming information from the neighboring vertices. Each bit node makes an independent final decision based on the cost function $\boldsymbol{\mu}$. This property allows the algorithm to be very fast but may cause the algorithm to output a word in $\mathbb{F}_2^n$ that is not a codeword.

2.4 Graph Cover Decoding

The local nature of message-passing iterative decoding described in the previous section prompts us to consider a *cover* of the Tanner graph.

Definition 2.9 *Let m be a positive integer and $H \in \mathbb{F}_2^{r\times n}$. A graph cover of the Tanner graph $T(H)$ of degree m is a bipartite graph $\widetilde{T}(H)$ such that for each vertex $v \in X \cup F$ there is a set of vertices $\{v_1, \ldots, v_m\}$ of $\widetilde{T}(H)$ with $\deg v_i = \deg v$ for all $1 \le i \le m$, and for every edge $\{u, v\}$ in $T(H)$ there are m edges from the vertices in $\{u_1, \ldots, u_m\}$ to the vertices in $\{v_1, \ldots, v_m\}$ connected in a 1-1 manner.*

We have seen that the parity-check matrix of a code $C(H)$ gives rise to a graph $T(H)$. Given a bipartite graph G, one can consider a matrix H and the code $C(H)$ such that the Tanner graph of $C(H)$ is G. Denote $\widetilde{C}(H)$ the code of length mn determined by $\widetilde{T}(H)$. Coordinates of a codeword of $\widetilde{C}(H)$ are ordered by successive

blocks of copies of each coordinate; we write a codeword $\tilde{\mathbf{c}}$ of $\widetilde{C}(H)$ as

$$(c_{(1,1)}, \ldots, c_{(1,m)}; \ldots; c_{(n,1)}, \ldots, c_{(n,m)}).$$

One may note that for a codeword $\mathbf{c} = (c_1, \ldots, c_n)$ of $C(H)$,

$$\mathbf{c}^{\uparrow m} := (c_1, \ldots, c_1; \ldots; c_n \ldots, c_n)$$

is a codeword of $\widetilde{C}(H)$.

For a codeword $\tilde{\mathbf{c}} = (c_{(1,1)}, \ldots, c_{(1,m)}; \ldots; c_{(n,1)}, \ldots, c_{(n,m)}) \in \widetilde{C}(H)$, the *projection* of $\tilde{\mathbf{c}}$ is a vector

$$\check{\tilde{\mathbf{c}}} = (\check{\tilde{c}}_1, \ldots, \check{\tilde{c}}_n) \in \mathbb{Z}^n \tag{2.2}$$

such that

$$\check{\tilde{c}}_i = \sum_{l=1}^{m} c_{(i,l)}, \tag{2.3}$$

and the *normalized projection* of $\tilde{\mathbf{c}}$ is a vector

$$\dot{\tilde{\mathbf{c}}} = (\dot{\tilde{c}}_1, \ldots, \dot{\tilde{c}}_n) \in [0,1]^n$$

where

$$\dot{\tilde{c}}_i = \frac{1}{m} \check{\tilde{c}}_i.$$

Example 2.10 *Consider again the code $C(H)$ where*

$$H = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix}.$$

This code was discussed in Examples 2.4 and 2.8. The Tanner graph $T(H)$ for this

code was shown in Figure 2.3. Here, Figure 2.4 illustrates a graph cover of $T(H)$ of degree 2 and a codeword

$$\tilde{\mathbf{c}} = (1, 0; 1, 1; 0, 1; 0, 0)$$

on $\widetilde{T}(H)$. Thus, we have $\check{\tilde{\mathbf{c}}} = (1, 2, 1, 0)$ and $\dot{\tilde{\mathbf{c}}} = \left(\frac{1}{2}, 1, \frac{1}{2}, 0\right)$. Recall from Example 2.4 that $\left(\frac{1}{2}, 1, \frac{1}{2}, 0\right)$ is also one of the vertices of $Q(H)$. We will see in Proposition 2.12 that this is not a coincidence.

The *graph cover decoder* finds the codeword in any finite degree graph cover that best explains the received vector $\mathbf{w}$. To make this precise, define

$$P(\tilde{\mathbf{w}}|\tilde{\mathbf{y}}) := \prod_{i=1}^{n} \prod_{k=1}^{m} P(w_{(i,k)}|y_{(i,k)}).$$

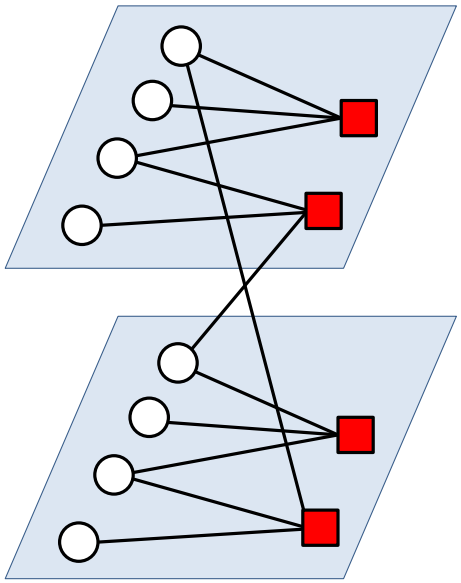
The graph cover decoder is the decision rule

$$\tilde{\mathbf{c}}_{GC} := \arg \max_{\tilde{\mathbf{y}} \in \widetilde{C}(H)} \frac{1}{m} \log \left(P\left(\mathbf{w}^{\uparrow m}|\tilde{\mathbf{y}}\right)\right).$$

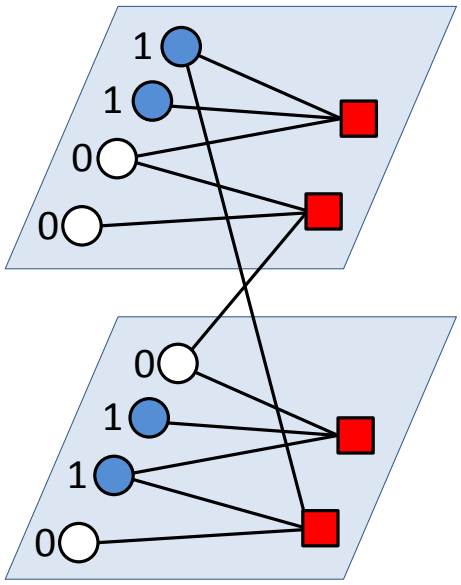
Note that graph cover decoder may not output a vector of length n. In fact, $\tilde{\mathbf{c}}_{GC}$ is a codeword of $\widetilde{C}(H)$ for some cover of degree m of $T(H)$. Thus, we focus instead on the graph cover pseudocodewords as defined below.

Definition 2.11 *Given an output $\tilde{\mathbf{c}}_{GC}$ from the graph cover decoder, the graph cover pseudocodeword is $\check{\tilde{\mathbf{c}}}_{GC}$, and normalized graph cover pseudocodeword is $\dot{\tilde{\mathbf{c}}}_{GC}$.*

The exact characterization of graph cover pseudocodewords and normalized graph cover pseudocodewords will be given in the next section.



Graph cover $\widetilde{T}(H)$



The codeword $(1,0;1,1;0,1;0,0)$ on $\widetilde{T}(H)$

Figure 2.4: A graph cover of $T(H)$ and a codeword of $\widetilde{C}(H)$ from Example 2.10

2.5 Pseudocodewords

Recall that given a code $C(H)$, $poly(C) \subseteq Q(H)$; the vertices of $Q(H)$ are called LP pseudocodewords; and the projected output from graph cover decoding are called graph cover pseudocodewords and normalized graph cover pseudocodewords. The following proposition provides a connection between LP pseudocodewords and graph cover pseudocodewords.

Proposition 2.12 *[40, Proposition 2.2] For a given received vector* $\mathbf{w}$*, let* $\mathbf{c}_{LP}$ *be the LP decoder decision, and let* $\dot{\tilde{\mathbf{c}}}_{GC}$ *be the normalized projection of the graph cover decoder decision. Then,*

$$\mathbf{c}_{LP} = \dot{\tilde{\mathbf{c}}}_{GC}.$$

Koetter and Vontobel [28] define an object called the *fundamental cone* which contains all graph cover pseudocodewords. In 2007, Koetter, Li, Vontobel, and Walker [27] are able to give an exact description of this cone.

Definition 2.13 *Given a parity-check matrix* $H \in \mathbb{F}_2^{r \times n}$*, the fundamental cone of* H *is defined as*

$$\mathcal{K}(H) := \left\{ \mathbf{v} \in \mathbb{R}^n \;\middle|\; v_i \geq 0 \text{ and } \sum_{l=1, l \neq i}^{n} h_{jl} v_l \geq h_{ji} v_i \text{ for all } 1 \leq i \leq n \text{ and } 1 \leq j \leq r \right\}.$$

In this same paper, graph cover pseudocodewords of $C(H)$ are characterized as those integer points within the fundamental cone $\mathcal{K}(H)$ which satisfy the parity-check conditions imposed by the rows of H. Their result may be stated as follows.

Theorem 2.14 *[27, Theorem 4.4] Let* $H \in \mathbb{F}_2^{r \times n}$*. Given* $\mathbf{p} \in \mathbb{Z}^n$*, the following are equivalent:*

1. $\mathbf{p}$ *is a graph cover pseudocodeword of the code* $C(H)$*;*

2. $\mathbf{p} \in \mathcal{K}(H)$ *and*

$$H\mathbf{p}^T = \mathbf{0} \mod 2. \tag{2.4}$$

According to Coleman [11], the fundamental cone $\mathcal{K}(H)$ is the conic hull of the relaxed polytope $Q(H)$. This unifies the notions of LP, graph cover, and normalized graph cover pseudocodeword. That is, every LP pseudocodeword and normalized graph cover pseudocodeword can be obtained by scaling graph cover pseudocodeword. Hence, we use the term *pseudocodeword* to refer to a graph cover pseudocodeword. Note here that every codeword is also a pseudocodeword.

Definition 2.15 *Given* $H \in \mathbb{F}_2^{r \times n}$*, let* $\mathcal{P}(H)$ *denote the set of pseudocodewords of* $C(H)$*.*

In the study of a mathematical object, it is natural to give special attention to the elements that are most "elementary". For example, understanding prime numbers is crucial to the study of natural numbers. Generators of a cyclic group and a basis of a vector space give us enough information to describe the entire algebraic structure. In our study of the pseudocodewords, we examine those that are *irreducible*.

Definition 2.16 *A nonzero pseudocodeword is said to be irreducible provided it cannot be written as a sum of two or more nonzero pseudocodewords. Given a parity-check matrix* $H \in \mathbb{F}_2^{r \times n}$*, the set of all irreducible pseudocodewords of* $C(H)$ *is denoted* $\mathcal{P}_{irr}(H)$*.*

It follows from the definition that any pseudocodeword can be written as a sum of irreducible pseudocodewords. Therefore, characterizing irreducible pseudocodewords is sufficient to describe the set of all pseudocodewords.

2.6 Cycle Codes

In this section, we introduce a class of codes called *cycle codes*. We will use the terminologies in accordance with [27, 37, 41], some of which may not be the standard convention in graph theory. For example, the term *cycle* used here is commonly referred to as *closed circuit*.

A linear code $C(H)$ is called a *cycle code* if all bit nodes in the associated Tanner graph $T(H)$ have degree 2. Recall that the vertex set of $T(H)$ can be written as $X \cup F$ where $X = \{x_1, \ldots, x_n\}$ corresponds to the bit nodes and $F = \{f_1, \ldots, f_r\}$ corresponds to the check nodes. For a cycle code $C(H)$, the *normal graph* of H, denoted $N(H)$, is formed by simply dropping the bit nodes from the Tanner graph $T(H)$. In other words, the normal graph of H is the graph with vertex set F and edge set $\{Nbhd(x) \mid x \in X\}$.

Example 2.17 *Consider*

$$H = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

as given in Example 2.7. Notice that each column of H has exactly two 1's. Equivalently, we can see from Figure 2.2 that all the bit nodes of the Tanner graph of the code $C(H)$ have degree 2. Therefore, $C(H)$ is a cycle code. The normal graph $N(H)$ is shown in Figure 2.5

A sequence of edges $(x_{i_1}, \ldots, x_{i_k})$ of $N(H)$ is called a *cycle* if the edges x_{i_1} can

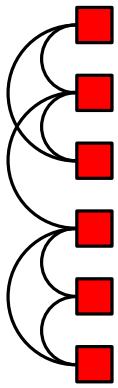


Figure 2.5: The normal graph $N(H)$ as described in Example 2.17

be directed so that x_{i_s} terminates where $x_{i_{s+1}}$ begins for all s where indices are taken modulo k. A cycle is called *simple* if each vertex of $N(H)$ is involved in at most two of the edges from $x_{i_1}, \ldots, x_{i_k}$. The term cycle code is derived from the following: the code $C(H)$ is precisely the code spanned by the characteristic vectors of the simple cycles in $N(H)$.

Let $\Gamma = (x_{i_1}, \ldots, x_{i_k})$ be a cycle in the normal graph $N(H)$. We say Γ is *tailless* if $x_{i_1} \neq x_{i_k}$ and Γ is *primitive* if there is no cycle Θ such that $\Gamma = \Theta^r$ with $r \geq 2$, i.e. Γ can be obtained by following Θ a total of r times. We say that the cycle $\Delta = (x_{j_1}, \ldots, x_{j_k})$ is equivalent to Γ if there exists an integer t such that $x_{i_s} = x_{j_{s+t}}$ for all s where indices are taken modulo k. The *monomial of* Γ is given by $g(\Gamma) := u_{i_1} \cdot \ldots \cdot u_{i_k}$ where the u_i's are indeterminants. Denote by $A(H)$ the collection of equivalence classes of tailless, primitive cycles in $N(H)$. Finally, the *edge zeta*

function of H [37] is the function

$$\zeta_H(u_1, \ldots, u_n) := \prod_{[\Gamma] \in A(H)} \frac{1}{1 - g(\Gamma)}$$

where $[\Gamma]$ denotes the equivalent class of Γ.

The following theorem, due to Koetter, Li, Vontobel, and Walker, describes the pseudocodewords of a cycle code via the edge zeta function of H.

Theorem 2.18 *[27, Theorem 5.9] Let $C(H)$ be a cycle code defined by a parity-check matrix H with a normal graph $N(H)$. The following are equivalent:*

1. *$u_1^{p_1} \cdots u_n^{p_n}$ has nonzero coefficient in ζ_H;*

2. *$(p_1, \ldots, p_n)$ is a pseudocodeword of $C(H)$.*

Since ζ_H is a rational function, the above theorem automatically implies that the the pseudocodewords of a cycle code correspond to the monomials appearing (with nonzero coefficient) in an expansion of a rational function.

Example 2.19 *Consider again the cycle code given by*

$$H = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

from Examples 2.7 and 2.17. Koetter, Li, Vontobel, and Walker [27] compute the

edge zeta function of H as

$$\begin{aligned}\zeta_H(u_1,\ldots,u_7) = \Big(&1 - 2u_1u_2u_3 + u_1^2u_2^2u_3^2 - 2u_5u_6u_7 + 4u_1u_2u_3u_5u_6u_7 \\ &- 2u_1^2u_2^2u_3^2u_5u_6u_7 - 4u_1u_2u_3u_4^2u_5u_6u_7 + 4u_1^2u_2^2u_3^2u_4^2u_5u_6u_7 \\ &+ u_5^2u_6^2u_7^2 - 2u_1u_2u_3u_5^2u_6^2u_7^2 + u_1^2u_2^2u_3^2u_5^2u_6^2u_7^2 \\ &+ 4u_1u_2u_3u_4^2u_5^2u_6^2u_7^2 - 4u_1^2u_2^2u_3^2u_4^2u_5^2u_6^2u_7^2\Big)^{-1}.\end{aligned}$$

We may obtain the first several terms of ζ_H by expanding out the Taylor series:

$$\begin{aligned}\zeta_H(u_1,\ldots,u_7) = 1 &+ 2u_1u_2u_3 + 3u_1^2u_2^2u_3^2 + 2u_5u_6u_7 + 4u_1u_2u_3u_5u_6u_7 \\ &+ 6u_1^2u_2^2u_3^2u_5u_6u_7 + 4u_1u_2u_3u_4^2u_5u_6u_7 + 12u_1^2u_2^2u_3^2u_4^2u_5u_6u_7 \\ &+ 3u_5^2u_6^2u_7^2 + 6u_1u_2u_3u_5^2u_6^2u_7^2 + 9u_1^2u_2^2u_3^2u_5^2u_6^2u_7^2 \\ &+ 12u_1u_2u_3u_4^2u_5^2u_6^2u_7^2 + 36u_1^2u_2^2u_3^2u_4^2u_5^2u_6^2u_7^2 + \ldots.\end{aligned}$$

It follows that the pseudocodewords of $C(H)$ are

$$\begin{aligned}&(0,0,0,0,0,0,0),(1,1,1,0,0,0,0),(2,2,2,0,0,0,0),(0,0,0,0,1,1,1),\\ &(1,1,1,0,1,1,1),(2,2,2,0,1,1,1),(1,1,1,2,1,1,1),(2,2,2,2,1,1,1),\\ &(0,0,0,0,2,2,2),(1,1,1,0,2,2,2),(2,2,2,0,2,2,2),(1,1,1,2,2,2,2),\\ &(2,2,2,2,2,2,2),\ldots.\end{aligned}$$

In [27], they also leave as an open problem to determine a rational function which encapsulates the pseudocodewords of a general parity-check code.

Chapter 3

Geometry Background

This chapter introduces standard terminologies from discrete geometry and several methods for enumerating integer points in a cone. Section 3.3 outlines the algorithm due to Barvinok which will be used to enumerate the pseudocodewords of a parity-check code in Chapter 4. Standard references for the material in this chapter are [2, 3, 4, 5, 36].

3.1 Algebra of Cones

A *rational polyhedron* $K \subset \mathbb{R}^d$ is the set of solutions of a finite system of linear inequalities with integer coefficients; that is,

$$K = \left\{ \mathbf{x} \in \mathbb{R}^d \mid \mathbf{c}_i \mathbf{x}^T \leq b_i, i = 1, \ldots, m \right\}$$

where $\mathbf{c}_i \in \mathbb{Z}^d$ and $b_i \in \mathbb{Z}$ for all i. A rational polyhedron $K \subset \mathbb{R}^d$ is called a *rational cone* if $\lambda \mathbf{v} \in K$ for all $\mathbf{v} \in K$ and $\lambda \geq 0$. Equivalently, K is a rational cone if and only if K can be defined as the set of solutions of a finite system of homogeneous

linear inequalities with integer coefficients; that is,

$$K = \left\{ \mathbf{x} \in \mathbb{R}^d \mid \mathbf{c}_i \mathbf{x}^T \leq 0, i = 1, \ldots, m \right\}$$

where $\mathbf{c}_i \in \mathbb{Z}^d$ for all i.

The set of *generators* of a rational cone K is any set of vectors $\{\mathbf{u}_1, \ldots, \mathbf{u}_k\}$ such that

$$K = \left\{ \sum_{i=1}^{k} \alpha_i \mathbf{u}_i \;\middle|\; \alpha_i \geq 0 \right\};$$

we call $\mathbf{u}_1, \ldots, \mathbf{u}_k$ generators of K. The generators $\mathbf{u}_1, \ldots, \mathbf{u}_k$ of K are *minimal* if for every set of generators $\mathbf{v}_1, \ldots, \mathbf{v}_l$ of K we have $k \leq l$. From now on, we say $\mathbf{u}_1, \ldots, \mathbf{u}_k$ are generators of K to mean that $\mathbf{u}_1, \ldots, \mathbf{u}_k$ are minimal generators of K, $\mathbf{u}_i \in \mathbb{Z}^d$, and $\mathbf{u}_i$ is not a multiple of an integer vector for all i. If $\text{span}\{\mathbf{u}_1, \ldots, \mathbf{u}_k\}$ is full-dimensional, then K is said to be *full-dimensional*; otherwise K is said to be *lower-dimensional*. If $\mathbf{u}_1, \ldots, \mathbf{u}_k$ are linearly independent, then K is said to be *simple*.

Definition 3.1 *A triangulation of a rational cone K is a finite set $\Gamma = \{S_1, \ldots, S_t\}$ of simple rational cones satisfying*

- $\cup_{i=1}^{t} S_i = K$,
- *every face of S_i is an element of Γ, and*
- *$S_i \cap S_j$ is a common face of S_i and S_j*

for all $1 \leq i \neq j \leq t$.

Proposition 3.2 *[36, Lemma 4.6.1] Given a rational cone K, there exists a triangulation $\Gamma = \{S_1, \ldots, S_t\}$ of K such that generators of S_i are among generators of K for all i.*

The *fundamental parallelepiped* of a rational cone K is the set

$$\Pi(K) := \left\{ \sum_{i=1}^{k} \alpha_i \mathbf{u}_i \;\middle|\; 0 \leq \alpha_i < 1 \right\}$$

where $\mathbf{u}_1, \ldots, \mathbf{u}_k$ are generators of K. The closure of $\Pi(K)$,

$$\overline{\Pi}(K) := \left\{ \sum_{i=1}^{k} \alpha_i \mathbf{u}_i \;\middle|\; 0 \leq \alpha_i \leq 1 \right\},$$

is called the *extended fundamental parallelepiped* of K. The *index* of a rational cone K, denoted $\mathrm{ind}(K)$, is the number of integer points in $\Pi(K)$; that is,

$$\mathrm{ind}(K) = \left| \left\{ \Pi(K) \cap \mathbb{Z}^d \right\} \right|.$$

If K is simple, the index of K is the same as the volume of $\Pi(K)$. A simple rational cone is said to be *unimodular* if it has index 1. Notice that if K is unimodular, then $\mathbf{0}$ is the unique integer point in $\Pi(K)$.

The set of integer vectors in a rational cone forms an additive semigroup whose minimal set of generators is called the *Hilbert basis* of the cone. More precisely, given a rational cone $K \subset \mathbb{R}^d$, the Hilbert basis of K is the minimal set of vectors $\{\mathbf{b}_1, \ldots, \mathbf{b}_t\}$ with the property that

$$\{\lambda_1 \mathbf{b}_1 + \ldots + \lambda_t \mathbf{b}_t \mid \lambda_1, \ldots, \lambda_t \geq 0 \text{ and } \lambda_1, \ldots, \lambda_t \in \mathbb{Z}\} = K \cap \mathbb{Z}^d.$$

The *shift* of a rational cone K by a vector $\mathbf{v}$ is the set

$$\mathbf{v} + K := \{\mathbf{v} + \mathbf{x} \mid \mathbf{x} \in K\}.$$

Notice that $\mathbf{v} + K$ is a cone that has a vertex at $\mathbf{v}$. The *dual* of a rational cone K is

$$K^* := \{\mathbf{x} \in \mathbb{R}^d \mid \mathbf{x}\mathbf{y}^T \geq 0 \ \forall \mathbf{y} \in K\}.$$

Hence, the dual of a rational cone is a rational cone. Another useful fact about the dual of a rational cone is given in the next proposition.

Proposition 3.3 (Bipolar Theorem) *Given a rational cone K, the dual of the dual of K is K itself; that is,*

$$(K^*)^* = K.$$

We are particularly interested in a function that represents every integer point in a rational cone. The following definition describes such function.

Definition 3.4 *Given a set $A \subseteq \mathbb{R}^d$, the generating function of A is*

$$f_A(\mathbf{x}) := \sum_{\mathbf{m} \in A \cap \mathbb{Z}^d} \mathbf{x}^{\mathbf{a}}$$

where

$$\mathbf{x}^{\mathbf{a}} = x_1^{a_1} x_2^{a_2} \cdots x_d^{a_d}.$$

If the set A contains a straight line with an infinite number of integer points, then we make the standard convention that $f_A(\mathbf{x}) \equiv 0$.

Notice that we use the term generating function to mean a possibly infinite series that records integer points in a set $A \subseteq \mathbb{R}^d$ as exponents of indeterminants.

Example 3.5 *Consider a half-open parallelepiped given by*

$$A = \left\{(x_1, x_2) \in \mathbb{R}^2 \mid 0 \leq 2x_1 - x_2 < 5 \text{ and } 0 \leq -x_1 + 3x_2 < 5\right\}.$$

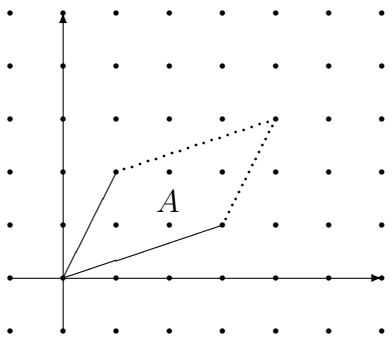


Figure 3.1: Parallelepiped A from Example 3.5

The parallelepiped A is shown in Figure 3.1. Then,

$$f_A(\mathbf{x}) = 1 + x_1x_2 + x_1^2x_2 + x_1^2x_2^2 + x_1^3x_2^2.$$

To facilitate our future discussion on the generating function of a rational cone, we define the *indicator function* of A as

$$\begin{array}{rccl} [A]: & \mathbb{R}^d & \to & \mathbb{R} \\ & \mathbf{m} & \mapsto & \begin{cases} 1 & \text{if } \mathbf{m} \in A \\ 0 & \text{if } \mathbf{m} \notin A. \end{cases} \end{array}$$

In other words, the indicator function "tests" membership of a set.

Generating functions of rational cones respect linear identities of their indicator functions as well as the indicator functions of their duals; more precisely, given rational cones $K_1, \ldots, K_t \subset \mathbb{R}^d$ and $\alpha_1, \ldots, \alpha_t \in \mathbb{Q}$,

$$\sum_{i=1}^{t} \alpha_i [K_i] = 0 \quad \Rightarrow \quad \sum_{i=1}^{t} \alpha_i f_{K_i}(\mathbf{x}) = 0 \tag{3.1}$$

and

$$\sum_{i=1}^{t} \alpha_i [K_i] = 0 \quad \Rightarrow \quad \sum_{i=1}^{t} \alpha_i [K_i^*] = 0. \tag{3.2}$$

These facts will be used in our study of generating functions of rational cones.

3.2 The Generating Function of a Cone

A function $f(x)$ is called *rational* if it can be written as a ratio of two polynomials, meaning

$$f(x) = \frac{p(x)}{q(x)}$$

where $p(x), q(x) \in \mathbb{R}[x]$ and $q(x) \neq 0$. We begin this section by stating the following well-known theorem.

Theorem 3.6 *[36, Theorem 4.6.11] The generating function of a rational cone is a rational function.*

We give a constructive proof for this theorem in Proposition 3.7 and Theorem 3.8 below.

Proposition 3.7 *[36, Corollary 4.6.8] For a simple rational cone K with generators $\mathbf{u}_1, \ldots, \mathbf{u}_k \in \mathbb{Z}^d$, we have*

$$f_K(\mathbf{x}) = \frac{f_{\Pi(K)}(\mathbf{x})}{(1 - \mathbf{x}^{\mathbf{u}_1}) \cdots (1 - \mathbf{x}^{\mathbf{u}_k})}. \tag{3.3}$$

Proof Every point $\mathbf{m} \in K \cap \mathbb{Z}^d$ can be written uniquely as

$$\mathbf{m} = \sum_{i=1}^{k} \alpha_i \mathbf{u}_i$$

where $\alpha_i \geq 0$ are real numbers for all i. Let $\lfloor \alpha \rfloor$ denote the largest integer not exceeding α (i.e., the integer part of α) and $\{\alpha\} = \alpha - \lfloor \alpha \rfloor$ (i.e., the fractional part of α). Then,

$$\mathbf{m} = \sum_{i=1}^{k} \lfloor \alpha_i \rfloor \mathbf{u}_i + \sum_{i=1}^{k} \{\alpha_i\} \mathbf{u}_i.$$

It is clear that $\sum_{i=1}^{k} \lfloor \alpha_i \rfloor \mathbf{u}_i$ is a non-negative integer combination of generators of K and $\sum_{i=1}^{k} \{\alpha_i\} \mathbf{u}_i$ is an integer point in $\Pi(K)$. ∎

Notice that the numerator of $f_K(\mathbf{x})$ in the expression given in (3.3) involves $\text{ind}(K)$ monomials. Therefore, Proposition 3.7 gives a rational form for the generating function of a simple rational cone. This can be extended to a rational cone as stated in the following theorem.

Theorem 3.8 *For a rational cone K with generators $\mathbf{u}_1, \ldots, \mathbf{u}_k \in \mathbb{Z}^d$, we have*

$$f_K(\mathbf{x}) = \frac{\sigma(\mathbf{x})}{(1 - \mathbf{x}^{\mathbf{u}_1}) \cdots (1 - \mathbf{x}^{\mathbf{u}_k})} \tag{3.4}$$

where $\sigma(\mathbf{x})$ is a polynomial.

Proof According to Proposition 3.2, K has a triangulation $\Gamma = \{S_1, \ldots, S_t\}$. Given $I \subseteq \{1, \ldots, t\}$, denote

$$S_I := \cap_{i \in I} S_i.$$

Applying inclusion-exclusion, we obtain

$$[K] = \sum_{j=1}^{t} (-1)^{j-1} \left(\sum_{\substack{I \subseteq \{1,\ldots,t\} \\ |I| = j}} [S_I] \right),$$

which implies that

$$f_K(\mathbf{x}) = \sum_{j=1}^{t} (-1)^{j-1} \left(\sum_{\substack{I \subseteq \{1,\ldots,t\} \\ |I|=j}} f_{S_I}(\mathbf{x}) \right). \tag{3.5}$$

According Definition 3.1, S_I is a simple rational cone. Moreover, generators of S_I are among generators of K. Applying Proposition 3.7 to (3.5), we obtain

$$f_K(\mathbf{x}) = \sum_{j=1}^{t} (-1)^{j-1} \left(\sum_{\substack{I \subseteq \{1,\ldots,t\} \\ |I|=j}} \frac{f_{\Pi(S_I)}(\mathbf{x})}{(1-\mathbf{x}^{\mathbf{u}_{I1}}) \cdots (1-\mathbf{x}^{\mathbf{u}_{Il_I}})} \right)$$

where $\mathbf{u}_{I1}, \ldots, \mathbf{u}_{Il_I}$ are generators of S_I. It follows that

$$f_K(\mathbf{x}) = \frac{\sigma(\mathbf{x})}{(1-\mathbf{x}^{\mathbf{u}_1}) \cdots (1-\mathbf{x}^{\mathbf{u}_k})}$$

where $\sigma(\mathbf{x})$ is a polynomial. ∎

3.3 Barvinok's Algorithm

While the theorems introduced in the previous section provide promising tools for constructing the generating function of integer points in a rational cone, the approach is still lacking in some sense. Namely, it involves enumerating all the integer points in the fundamental parallelepipeds $\Pi(S_I)$ and direct application of inclusion-exclusion, both of which can be quite costly. This section outlines a more efficient method due to Barvinok.

Introduced in 1994, Barvinok's algorithm is a polynomial time algorithm for counting the number of lattice points in a convex polyhedron in a fixed dimension

[2]. The algorithm has seen applications in optimization, statistics, and algebra; we present the aspects of the algorithms which are relevant to the enumeration of integer points in a rational cone. Full treatment of the algorithm, as well as surveys, applications, and several improvements can be found in [2, 3, 4, 6, 20, 23, 24].

Barvinok's algorithm was inspired by the fact that a long polynomial or infinite series can often be rewritten as a much shorter rational function. For example,

$$\sum_{m=0}^{\infty} x^m = \frac{1}{1-x}.$$

While $\sum_{m=0}^{\infty} x^m$ involves an infinite number of monomials, it can be written as $\frac{1}{1-x}$ which involves only 3 monomials.

Equation (3.3) suggests that the generating functions of cones of smaller indices involve fewer monomials. The following lemma provides an efficient method for decomposing a simple rational cone into simple rational cones of smaller indices.

Lemma 3.9 *[2, Theorem 5.4] Fix d. Given a full-dimensional simple rational cone $K \subset \mathbb{R}^d$, there exists an algorithm that computes a signed decomposition*

$$[K] = \sum_{i \in I} \epsilon_i [K_i] + \sum_{j \in J} \epsilon_j [S_j]$$

where $\epsilon_i, \epsilon_j \in \{-1, 1\}$, K_i is a full-dimensional simple rational cone, $\operatorname{ind}(K_i) < \operatorname{ind}(K)$*, and S_j is a lower-dimensional rational cone for all $i \in I$ and $j \in J$.*

Proof Let $\mathbf{u}_1, \ldots, \mathbf{u}_d \in \mathbb{Z}^d$ be generators of K. Consider a parallelepiped

$$A = \left\{ \sum_{i=1}^{d} \alpha_i \mathbf{u}_i \;\middle|\; |\alpha_i| \leq \Big(\operatorname{ind}(K)\Big)^{-\frac{1}{d}} \right\}.$$

We observe that A is symmetric about the origin and the volume of A is 2^d. By

Minkowski's Convex Body Theorem, there is a nonzero integer point $\mathbf{v} \in A$. For $i = 1, \ldots, d$, let K_i be the cone generated by $\mathbf{u}_1, \ldots, \mathbf{u}_{i-1}, \mathbf{v}, \mathbf{u}_{i+1}, \ldots, \mathbf{u}_d$. Then,

$$\mathrm{ind}(K_i) \leq \Big(\mathrm{ind}(K)\Big)^{\frac{d-1}{d}}$$

and there exists a decomposition

$$[K] = \sum_{i \in \{1,2,\ldots,d\}} \epsilon_i [K_i] + \sum_{j \in J} \epsilon_j [S_j]$$

where S_j ranges over lower-dimensional faces of K_i, and $\epsilon_i, \epsilon_j \in \{-1, 1\}$. ∎

The following lemma is known as Brion's polarization trick [10].

Lemma 3.10 *If $K \subset \mathbb{R}^d$ is a rational cone such that K^* is a lower-dimensional simple rational cone, then*

$$f_K(\mathbf{x}) = 0.$$

Proof Since K^* is a lower-dimensional simple rational cone, K is a cone that contains a straight line. It follows that $f_K(\mathbf{x}) = 0$ from the standard convention given in Definition 3.4. ∎

Theorem 3.11 *Fix d. Given a rational cone $K \subset \mathbb{R}^d$, there exists an algorithm that computes the generating function of K as a finite sum*

$$f_K(\mathbf{x}) = \sum_i \epsilon_i \frac{1}{(1 - \mathbf{x}^{\mathbf{v}_{i1}}) \cdots (1 - \mathbf{x}^{\mathbf{v}_{ik}})}$$

where $\epsilon_i \in \{-1, 1\}$ and $\mathbf{v}_{ij}$ are integer vectors for all i, j.

Proof First, triangulate the dual of K to obtain

$$[K^*] = \sum_{i \in I} [K_i^*] + \sum_{j \in J} [S_j^*]$$

where K_i^* is full-dimensional and S_j^* is lower-dimensional for all i, j. We may apply (3.1) and (3.2) and see that

$$[K] = \sum_{i \in I} [K_i] + \sum_{j \in J} [S_j]$$

and

$$f_K(\mathbf{x}) = \sum_{i \in I} f_{K_i}(\mathbf{x}) + \sum_{j \in J} f_{S_j}(\mathbf{x}).$$

Then Lemma 3.10 yields

$$f_K(\mathbf{x}) = \sum_{i \in I} f_{K_i}(\mathbf{x})$$

Hence, the cones S_j^*, $j \in J$, can be safely discarded with no effect on the generating function of K.

Next, apply the decomposition of Lemma 3.9 to each K_i^*. For $i \in I$, this gives

$$[K_i^*] = \sum_{l \in I_i} \epsilon_{il} [K_{il}^*] + \sum_{j \in J_i} \epsilon_{ij} [S_{ij}^*]$$

where the index of K_{il}^* is less than that of K_i^* and S_{ij}^* is lower-dimensional for all l, j. Thus, we may once again discard the cones S_{ij}^*, $j \in J_i$. We iterate this procedure, applying Lemma 3.9 next to the K_{il}^* with $l \in I_i$. Each time, the indices of the simple rational cones obtained decrease (doubly exponentially [3]). Continue until a

decomposition

$$[K_i^*] = \sum_{l \in I_i} \epsilon_{il} [U_{il}^*] \mod \text{indicators of lower-dimensional cones}$$

is obtained with all U_{ij}^* unimodular.

We may now write the dual of K as

$$[K^*] = \sum_{i \in I} \sum_{l \in I_i} \epsilon_{il} [U_{il}^*] \mod \text{indicators of lower-dimensional cones.}$$

Apply Equation (3.2) and Lemma 3.10 to obtain

$$f_K(\mathbf{x}) = \sum_{i \in I} \sum_{l \in I_i} \epsilon_{il} f_{U_{il}}(\mathbf{x}).$$

Since U_{il} is unimodular for all i and $l \in I_i$ [3], we have

$$f_{U_{il}}(\mathbf{x}) = \frac{1}{(1 - \mathbf{x}^{\mathbf{v}_{il1}}) \cdots (1 - \mathbf{x}^{\mathbf{v}_{ilk}})}$$

where $\mathbf{v}_{il1}, \ldots, \mathbf{v}_{ilk}$ are generators of U_{il}. This completes the proof of the theorem. ∎

The procedure detailed in the proof of Theorem 3.11 is known as Barvinok's algorithm. The complexity of this algorithm is $\mathcal{L}^{O(d)}$ where $\mathcal{L}$ is the input size of K. For a rational cone K given by

$$K = \left\{ \mathbf{x} \in \mathbb{R}^d \mid \mathbf{c}_i \mathbf{x}^T \leq 0, i = 1, \ldots, m \right\},$$

the input size of K is

$$\mathcal{L} = \sum_{i=1}^{m} \sum_{j=1}^{d} \chi(c_{i_j}).$$

Barvinok's algorithm has been implemented in several software packages such as Barvinok 0.27 [39], LattE [20], and LattE macchiato [22].

Example 3.12 *Let* $\mathbf{u}_1 = (1,2), \mathbf{u}_2 = (3,1) \in \mathbb{Z}^2$. *The simple rational cone* K *whose generators are* $\mathbf{u}_1$ *and* $\mathbf{u}_2$ *and its fundamental parallelepiped are shown in Figure 3.2. Note also that* $\Pi(K)$ *is the half-open parallelepiped discussed in Example 3.5.*

Then, according to Proposition 3.7, the generating function of K *can be given by*

$$\begin{aligned} f_K(\mathbf{x}) &= \frac{f_{\Pi(K)}(\mathbf{x})}{(1-\mathbf{x}^{\mathbf{u}_1})(1-\mathbf{x}^{\mathbf{u}_2})} \\ &= \frac{1 + x_1x_2 + x_1^2x_2 + x_1^2x_2^2 + x_1^3x_2^2}{(1-x_1x_2^2)(1-x_1^3x_2)}. \end{aligned}$$

However, Barvinok's algorithm, as executed by Barvinok 0.27 [39], yields

$$f_K(\mathbf{x}) = \frac{1}{(1-x_1)(1-x_2)} - \frac{x_1}{(1-x_1)(1-x_1^3x_2)} - \frac{x_2}{(1-x_2)(1-x_1x_2^2)}.$$

The above rational form results from noting that the integer points in K *are the same as those in*

$$K_1 \setminus (K_2 \cup K_3)$$

where K_1 *is the first quadrant,* K_2 *is the shift of the rational cone generated by* $(1,0)$ *and* $(3,1)$ *by the vector* $(1,0)$*, and* K_3 *is the shift of the rational cone generated by* $(0,1)$ *and* $(1,2)$ *by the vector* $(0,1)$ *as shown in Figure 3.3. Here, note that* K_1*,* K_2*, and* K_3 *are all shifts of unimodular cones.*

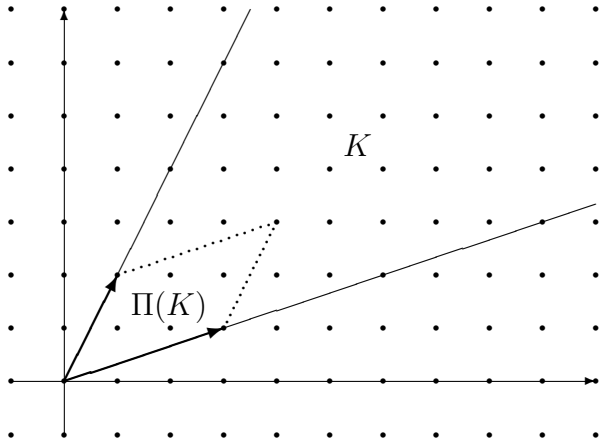


Figure 3.2: The cone K and its fundamental parallelepiped $\Pi(K)$ given in Example 3.12

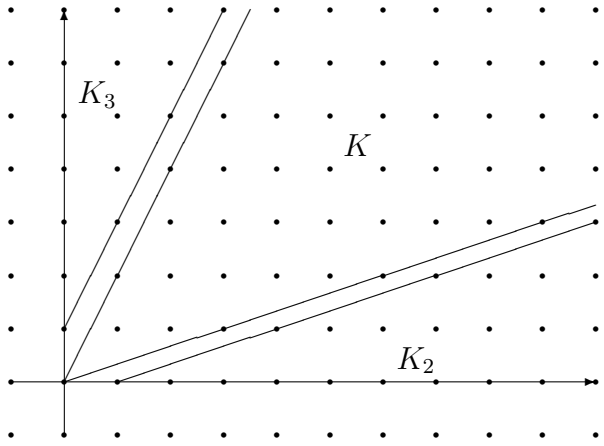


Figure 3.3: The cone K from Example 3.12 as being treated by Barvinok's algorithm

Chapter 4

Lifting the Fundamental Cone and Enumerating Pseudocodewords

In this chapter, we define the lifted fundamental cone of a parity-check code and use the ingredients from Chapters 2 and 3 to enumerate the pseudocodewords of a binary linear code. In particular, Section 4.1 gives the definition and basic properties of the lifted fundamental cone. Section 4.2 discusses monomial substitution of a rational function and the generating function of the pseudocodewords. Finally, minimal and irreducible pseudocodewords are introduced in Section 4.3.

4.1 The Lifted Fundamental Cone

Recall that given a parity-check matrix H, the fundamental cone $\mathcal{K}(H)$ is a rational cone that contains all the pseudocodewords of $C(H)$. This motivates us to apply the tools described in Chapter 3, in particular Barvinok's algorithm, to enumerate the pseudocodewords of $C(H)$. However, as we note from Theorem 2.14, not every integer point of $\mathcal{K}(H)$ is a pseudocodeword of $C(H)$. While applying

Barvinok's algorithm to $\mathcal{K}(H)$ will produce a list of integer points that contains all pseudocodewords, this list will likely contain words that are not pseudocodewords. To circumvent this, we introduce a rational cone $\hat{\mathcal{K}}(H)$ such that the integer points of $\hat{\mathcal{K}}(H)$ correspond precisely to the pseudocodewords of $C(H)$. Hence, this object allows us to apply techniques developed in the previous chapter.

Definition 4.1 *Given a parity-check matrix $H \in \mathbb{F}_2^{r\times n}$, the lifted fundamental cone of H is*

$$\hat{\mathcal{K}}(H) = \left\{ (\mathbf{v},\mathbf{a}) \in \mathbb{R}^{n+r} \;\middle|\; \begin{array}{l} v_i \geq 0, H\mathbf{v}^T = 2\mathbf{a}^T, \text{ and} \\ \sum_{l=1}^{n} h_{jl}v_l \geq 2h_{ji}v_i \text{ for all } 1 \leq i \leq n \text{ and } 1 \leq j \leq r \end{array} \right\}.$$

It turns out that the use of the vector $\mathbf{a}$ in Definition 4.1 is similar to that in the reformulation of ML decoding as an integer programming problem in [9].

To relate the lifted cone $\hat{\mathcal{K}}(H)$ to the fundamental cone $\mathcal{K}(H)$, define the projection

$$\begin{array}{rccc} \pi: & \mathbb{R}^{n+r} & \to & \mathbb{R}^n \\ & (\mathbf{v},\mathbf{a}) & \mapsto & \mathbf{v}. \end{array} \tag{4.1}$$

We make the relationship between the lifted fundamental cone $\hat{\mathcal{K}}(H)$ and the pseudocodewords of $C(H)$ precise in the following proposition.

Proposition 4.2 *Let $H \in \mathbb{F}_2^{r\times n}$. The projection $\pi|_{\hat{\mathcal{K}}(H)}$ is one-to-one and*

$$\pi\left(\hat{\mathcal{K}}(H)\right) = \mathcal{K}(H).$$

Furthermore,

$$\pi\left(\hat{\mathcal{K}}(H) \cap \mathbb{Z}^{n+r}\right) = \mathcal{P}(H).$$

In other words, $\hat{\mathcal{K}}(H)$ is a cone in $\mathbb{R}^{n+r}$ whose projection is $\mathcal{K}(H)$, and its integer points correspond precisely to the pseudocodewords of $C(H)$.

Proof Suppose that $\pi(\mathbf{v}, \mathbf{a}) = \pi(\mathbf{v}', \mathbf{a}')$ where $(\mathbf{v}, \mathbf{a}), (\mathbf{v}', \mathbf{a}') \in \hat{\mathcal{K}}(H)$. Then, $\mathbf{v} = \mathbf{v}'$ and

$$2\mathbf{a}^T = H\mathbf{v}^T = H\mathbf{v}'^T = 2\mathbf{a}'^T.$$

We conclude that $(\mathbf{v}, \mathbf{a}) = (\mathbf{v}', \mathbf{a}')$ and $\pi|_{\hat{\mathcal{K}}(H)}$ is injective.

Now,

$$\begin{aligned}
\pi\left(\hat{\mathcal{K}}(H)\right) &= \left\{ \mathbf{v} \in \mathbb{R}^n \;\middle|\; \begin{array}{l} v_i \geq 0, H\mathbf{v}^T = 2\mathbf{a}^T \text{ for some } \mathbf{a} \in \mathbb{R}^r, \text{ and} \\ \sum_{l=1}^{n} h_{jl} v_l \geq 2h_{ji} v_i \text{ for all } 1 \leq i \leq n \text{ and } 1 \leq j \leq r \end{array} \right\} \\
&= \left\{ \mathbf{v} \in \mathbb{R}^n \;\middle|\; \begin{array}{l} v_i \geq 0, \text{ and} \\ \sum_{l=1}^{n} h_{jl} v_l \geq 2h_{ji} v_i \text{ for all } 1 \leq i \leq n \text{ and } 1 \leq j \leq r \end{array} \right\} \\
&= \mathcal{K}(H)
\end{aligned}$$

where the last equality follows from Definition 2.13.

Let $(\mathbf{v}, \mathbf{a})$ be an integer point in $\hat{\mathcal{K}}(H)$. Then,

$$\mathbf{v} = \pi(\mathbf{v}, \mathbf{a}) \in \mathcal{K}(H)$$

and

$$H\mathbf{v}^T = \mathbf{0} \mod 2$$

since $H\mathbf{v}^T = 2\mathbf{a}^T$. It follows from Theorem 2.14 that $\pi(\mathbf{v}, \mathbf{a}) = \mathbf{v}$ is a pseudocodeword of $C(H)$. On the other hand, let $\mathbf{p} \in \mathcal{P}(H)$. Then, $\mathbf{p}$ is an integer point in $\mathcal{K}(H)$ such that $H\mathbf{p}^T = \mathbf{0} \mod 2$. Since $\pi\left(\hat{\mathcal{K}}(H)\right) = \mathcal{K}(H)$, $(\mathbf{p}, \mathbf{a}) \in \hat{\mathcal{K}}(H)$ for some $\mathbf{a} \in \mathbb{R}^r$. However, as $H\mathbf{p}^T = \mathbf{0} \mod 2$, $\mathbf{a}$ must be an integer vector. It follows that $(\mathbf{p}, \mathbf{a})$ is

an integer point in $\hat{\mathcal{K}}(H)$. We now conclude that $\pi\left(\hat{\mathcal{K}}(H) \cap \mathbb{Z}^{n+r}\right) = \mathcal{P}(H)$. ∎

4.2 Enumerating Pseudocodewords

While the results developed in Chapter 3 apply directly to the rational cone $\hat{\mathcal{K}}(H)$, we are more interested in the projection of $\hat{\mathcal{K}}(H)$. Specifically, Proposition 4.2 suggests that the generating function for the pseudocodewords can be obtained by specializing $f_{\hat{\mathcal{K}}(H)}(\mathbf{x})$. The method of monomial substitution due to Barvinok and Woods in the following lemma provides the details for this task.

Lemma 4.3 *[4, Theorem 2.6] Given a rational function*

$$f(\mathbf{x}) = \sum_{i \in I} \alpha_i \frac{\mathbf{x}^{\mathbf{a}_i}}{(1 - \mathbf{x}^{\mathbf{u}_{i1}}) \cdots (1 - \mathbf{x}^{\mathbf{u}_{is}})} \tag{4.2}$$

where I is a finite set, $\alpha_i \in \mathbb{Q}$, $\mathbf{a}_i, \mathbf{u}_{ij} \in \mathbb{Z}^n$, and $\mathbf{u}_{ij} \neq \mathbf{0}$ for all i, j, and a monomial map

$$\begin{array}{rccc} \phi: & \mathbb{C}^d & \rightarrow & \mathbb{C}^n \\ & \mathbf{z} & \mapsto & \left(\mathbf{z}^{\mathbf{l}_1}, \ldots, \mathbf{z}^{\mathbf{l}_n}\right) \end{array} \tag{4.3}$$

where $\mathbf{l}_1, \ldots, \mathbf{l}_n \in \mathbb{Z}^d$ such that the image of ϕ does not lie entirely in the set of poles of $f(\mathbf{x})$, there is an algorithm which computes $f(\phi(\mathbf{z}))$ in the form

$$f(\phi(\mathbf{z})) = \sum_{i \in I'} \beta_i \frac{\mathbf{z}^{\mathbf{b}_i}}{(1 - \mathbf{z}^{\mathbf{w}_{i1}}) \cdots (1 - \mathbf{z}^{\mathbf{w}_{it}})}$$

where $t \leq s$, I' is a finite set, $\beta_i \in \mathbb{Q}$, $\mathbf{b}_i, \mathbf{w}_{ij} \in \mathbb{Z}^d$, and $\mathbf{w}_{ij} \neq \mathbf{0}$ for all i, j.

Proof For $\mathbf{c} \in \mathbb{C}^n$ with $\mathbf{c} = \mathbf{c}_1 + i\mathbf{c}_2$ where $\mathbf{c}_1, \mathbf{c}_2 \in \mathbb{R}^n$ and $\mathbf{a} \in \mathbb{R}^n$, let

$$\langle \mathbf{c}, \mathbf{a} \rangle = \langle \mathbf{c_1}, \mathbf{a} \rangle + i \langle \mathbf{c_2}, \mathbf{a} \rangle$$

where $\langle \cdot, \cdot \rangle$ is the standard scalar product in $\mathbb{R}^n$. We associate with the rational function (4.2) a meromorphic function $F(\mathbf{c})$, $\mathbf{c} \in \mathbb{C}^n$, defined by

$$F(\mathbf{c}) = \sum_{i \in I} \alpha_i \frac{\exp\langle \mathbf{c}, \mathbf{a}_i \rangle}{(1 - \exp\langle \mathbf{c}, \mathbf{u}_{i1} \rangle) \cdots (1 - \exp\langle \mathbf{c}, \mathbf{u}_{is} \rangle)}. \tag{4.4}$$

Notice that the set of poles of the i^{th} fraction of (4.4) is

$$\bigcup_{j=1}^{s} \left\{ \mathbf{c} \in \mathbb{C}^d \mid \langle \mathbf{c}, \mathbf{u}_{ij} \rangle = 0 \right\}$$

and the set of poles of $F(\mathbf{c})$ could be much smaller because of cancellations of singularities.

For $\mathbf{c} = (\gamma_1, \ldots, \gamma_n)$ and $\mathbf{x} = (x_1, \ldots, x_n)$ with $x_i = \exp(\gamma_i)$ for all $i = 1, \ldots, n$, we write

$$\mathbf{x} = \exp(\mathbf{c}).$$

Then the functions (4.2) and (4.4) are related by the identity

$$F(\mathbf{c}) = f(\exp(\mathbf{c})).$$

With the monomial map (4.3) we associate a linear transformation

$$\begin{array}{rccc} \Phi: & \mathbb{C}^d & \to & \mathbb{C}^n \\ & \mathbf{c} & \mapsto & (\langle \mathbf{c}, \mathbf{l}_1 \rangle, \ldots, \langle \mathbf{c}, \mathbf{l}_n \rangle) \end{array}$$

and the adjoint transformation

$$\begin{array}{rccc} \Phi^*: & \mathbb{C}^n & \to & \mathbb{C}^d \\ & \boldsymbol{\xi} & \mapsto & \xi_1 \mathbf{l}_1 + \ldots + \xi_n \mathbf{l}_n. \end{array}$$

For $\mathbf{c} \in \mathbb{C}^d$, Define

$$g(\mathbf{z}) = f(\phi(\mathbf{z}))$$

and

$$G(\mathbf{c}) = F(\Phi(\mathbf{c})).$$

Hence,

$$G(\mathbf{c}) = g(\exp(\mathbf{c})).$$

Let $L \subseteq \mathbb{C}^n$ be the image of $\mathbb{C}^d$ under Φ; that is, $L := \Phi(\mathbb{C}^d)$. Then, L does not lie entirely in the set of poles off $F(\mathbf{c})$. We can then compute $G(\mathbf{c})$ in the form

$$G(\mathbf{c}) = \sum_{i \in I'} \beta_i \frac{\exp\langle \Phi(\mathbf{c}), \mathbf{q}_i \rangle}{(1 - \exp\langle \Phi(\mathbf{c}), \mathbf{v}_{i1} \rangle) \cdots (1 - \exp\langle \Phi(\mathbf{c}), \mathbf{v}_{is} \rangle)}$$

where $\beta_i \in \mathbb{Q}$, $\mathbf{q}_i, \mathbf{v}_{ij} \in \mathbb{Z}^d$, and $\langle \Phi(\mathbf{c}), \mathbf{v}_{is} \rangle \neq 0$ for a generic $\mathbf{c} \in L$ for all i, j. We now let

$$\mathbf{b}_i = \Phi^*(\mathbf{q}_i)$$

and

$$\mathbf{w}_{ij} = \Phi^*(\mathbf{v}_{ij})$$

so that

$$G(\mathbf{c}) = \sum_{i \in I'} \beta_i \frac{\exp\langle \mathbf{c}, \mathbf{b}_i \rangle}{(1 - \exp\langle \mathbf{c}, \mathbf{w}_{i1} \rangle) \cdots (1 - \exp\langle \mathbf{c}, \mathbf{w}_{is} \rangle)}.$$

The result follows since $g(\exp(\mathbf{c})) = G(\mathbf{c})$. ∎

We are now equipped to prove that the generating function for the pseudocodewords of a general parity-check code is a rational function. In fact, we will give two distinctive proofs: Theorem 4.4 specializes the generating function of the lifted fundamental cone and Theorem 4.5 uses Barvinok's algorithm and the monomial substi-

tution method given in Lemma 4.3. Both approaches involve the lifted fundamental cone and give a specific and often different rational form for the generating function of the pseudocodewords.

Theorem 4.4 *Given a parity-check matrix $H \in \mathbb{F}_2^{r\times n}$, the generating function of the pseudocodewords of $C(H)$ may be expressed as*

$$f_{\mathcal{P}(H)}(\mathbf{x}) = \frac{\sigma(\mathbf{x})}{(1-\mathbf{x}^{\mathbf{v}_1})\cdots(1-\mathbf{x}^{\mathbf{v}_l})}$$

where $\sigma(\mathbf{x})$ is a polynomial and $\mathbf{v}_1, \ldots, \mathbf{v}_l$ are integer vectors. Furthermore, for $i = 1, \ldots, l$,

$$\mathbf{v}_i = \pi(\mathbf{u}_i)$$

where π is the projection given in (4.1) *and $\mathbf{u}_1, \ldots, \mathbf{u}_l$ are generators of the lifted fundamental cone $\hat{\mathcal{K}}(H)$.*

Proof For $\mathbf{x} \in \mathbb{C}^{n+r}$, consider the generating function of the lifted fundamental cone

$$f_{\hat{\mathcal{K}}(H)}(x_1, x_2, \ldots, x_{n+r}) = \sum_{(\mathbf{v},\mathbf{a})\in\hat{\mathcal{K}}(H)\cap\mathbb{Z}^{n+r}} \mathbf{x}^{(\mathbf{v},\mathbf{a})}.$$

If $(x_1, \ldots, x_n, 1, \ldots, 1)$ is not a pole of the expression of $f_{\hat{\mathcal{K}}(H)}(\mathbf{x})$, then

$$\begin{aligned} f_{\hat{\mathcal{K}}(H)}(x_1, \ldots, x_n, 1, \ldots, 1) &= \sum_{(\mathbf{v},\mathbf{a})\in\hat{\mathcal{K}}(H)\cap\mathbb{Z}^{n+r}} \mathbf{x}^{\mathbf{v}} \\ &= \sum_{\mathbf{v}\in\pi(\hat{\mathcal{K}}(H)\cap\mathbb{Z}^{n+r})} \mathbf{x}^{\mathbf{v}} \\ &= \sum_{\mathbf{v}\in\mathcal{P}(H)} \mathbf{x}^{\mathbf{v}} \end{aligned}$$

where the last equality follows from Proposition 4.2.

Now, the cone $\hat{\mathcal{K}}(H)$ is clearly a rational cone, and by Theorem 3.8 its generating function can be written as

$$f_{\hat{\mathcal{K}}(H)}(\mathbf{x}) = \frac{\sigma(\mathbf{x})}{(1-\mathbf{x}^{\mathbf{u}_1})\cdots(1-\mathbf{x}^{\mathbf{u}_l})} \tag{4.5}$$

where $\sigma(\mathbf{x})$ is a polynomial and $\mathbf{u}_1, \ldots, \mathbf{u}_l$ are generators of $\hat{\mathcal{K}}(H)$.

If $(\mathbf{0}, \mathbf{a}) \in \hat{\mathcal{K}}(H)$, then

$$H\mathbf{0}^T = 2\mathbf{a}^T,$$

implying that $\mathbf{a} = \mathbf{0}$. Thus, $(\mathbf{0}, \mathbf{a})$ cannot be among generators of $\hat{\mathcal{K}}(H)$ and $(x_1, \ldots, x_n, 1, \ldots, 1)$ is not a pole of the expression of $f_{\hat{\mathcal{K}}(H)}(\mathbf{x})$ given in (4.5). We conclude that

$$\begin{aligned} f_{\mathcal{P}(H)}(\mathbf{x}) &= f_{\hat{\mathcal{K}}(H)}(x_1, \ldots, x_n, 1, \ldots, 1) \\ &= \frac{\sigma(\mathbf{x})}{(1-\mathbf{x}^{\mathbf{v}_1})\cdots(1-\mathbf{x}^{\mathbf{v}_l})} \end{aligned}$$

where $\sigma(\mathbf{x})$ is a polynomial and $\mathbf{v}_i = \pi(\mathbf{u}_i)$ for all i. ∎

Theorem 4.5 *Fix $d := n + r$. Given a parity-check matrix $H \in \mathbb{F}_2^{r \times n}$, there exists an algorithm which computes the generating function of the pseudocodewords of $C(H)$ as a finite sum*

$$f_{\mathcal{P}(H)}(\mathbf{x}) = \sum_i \epsilon_i \frac{\mathbf{x}^{\mathbf{a}_i}}{(1-\mathbf{x}^{\mathbf{v}_{i1}})\cdots(1-\mathbf{x}^{\mathbf{v}_{id}})}$$

where $\epsilon_i \in \mathbb{Q}$ and $\mathbf{a}_i$, $\mathbf{v}_{ij}$ are integer vectors for all i, j.

Proof We apply Theorem 3.11 to the lifted fundamental cone and obtain

$$f_{\hat{\mathcal{K}}(H)}(\mathbf{x}) = \sum_i \epsilon_i \frac{1}{(1-\mathbf{x}^{\mathbf{u}_{i1}})\cdots(1-\mathbf{x}^{\mathbf{u}_{ik}})} \tag{4.6}$$

where $\epsilon_i \in \{-1, 1\}$ and $\mathbf{u}_{ij}$ are integer vectors for all i, j.

Recall that $\mathbf{e}_i = (0, \ldots, 0, 1, 0, \ldots, 0) \in \mathbb{Z}^{n+r}$ is the vector with 1 in the i^{th} coordinate and 0's elsewhere. Consider the monomial map

$$\begin{array}{rccc} \phi: & \mathbb{C}^n & \to & \mathbb{C}^{n+r} \\ & \mathbf{z} & \mapsto & \left(\mathbf{z}^{\mathbf{l}_1}, \ldots, \mathbf{z}^{\mathbf{l}_{n+r}}\right) \end{array} \tag{4.7}$$

where

$$\mathbf{l}_i = \begin{cases} \mathbf{e}_i & \text{if } 1 \leq i \leq n \\ \mathbf{0} & \text{if } n+1 \leq i \leq n+r. \end{cases}$$

Then,

$$\phi(z_1, z_2, \ldots, z_n) = (z_1, \ldots, z_n, 1, \ldots, 1)$$

and the image of ϕ does not lie entirely in the set of poles of $f_{\hat{\mathcal{K}}(H)}(\mathbf{x})$.

Now, similar to the proof of Theorem 4.4, we have

$$f_{\mathcal{P}(H)}(\mathbf{z}) = f_{\hat{\mathcal{K}}(H)}(\phi(\mathbf{z})).$$

Applying Lemma 4.3 to (4.6) with the monomial map (4.7) finishes the proof of this theorem. ∎

Example 4.6 *Consider the code $C(H)$ from Example 2.4 given by*

$$H = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix}.$$

We would like to find a rational form for the generating function of $\mathcal{P}(H)$ using the approaches of Theorem 4.4 and Theorem 4.5. We will explicitly illustrate the procedures from Theorem 4.4. The routines used in Theorem 4.5 will be carried out

by Barvinok 0.27 [20].

Generators of the lifted fundamental cone $\hat{\mathcal{K}}(H)$ *are*

$$\mathbf{u}_1 = (0, 1, 1, 1, 1, 1),$$
$$\mathbf{u}_2 = (1, 0, 1, 0, 1, 1),$$
$$\mathbf{u}_3 = (1, 1, 0, 1, 1, 1),$$
$$\mathbf{u}_4 = (1, 0, 1, 2, 1, 2), \text{ and}$$
$$\mathbf{u}_5 = (1, 2, 1, 0, 2, 1).$$

Consider a triangulation

$$[\hat{\mathcal{K}}(H)] = [S_1] + [S_2] - [S_3]$$

where

S_1 *is the cone generated by* $\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3$, *and* $\mathbf{u}_4$,

S_2 *is the cone generated by* $\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3$, *and* $\mathbf{u}_5$,

and

$$S_3 = S_1 \cap S_2.$$

We can then compute

$$\begin{aligned} f_{\hat{\mathcal{K}}(H)}(\mathbf{x}) &= f_{S_1}(\mathbf{x}) + f_{S_2}(\mathbf{x}) - f_{S_3}(\mathbf{x}) \\ &= \frac{1}{(1-\mathbf{x}^{\mathbf{u}_1})(1-\mathbf{x}^{\mathbf{u}_2})(1-\mathbf{x}^{\mathbf{u}_3})(1-\mathbf{x}^{\mathbf{u}_4})} \\ &\quad + \frac{1}{(1-\mathbf{x}^{\mathbf{u}_1})(1-\mathbf{x}^{\mathbf{u}_2})(1-\mathbf{x}^{\mathbf{u}_3})(1-\mathbf{x}^{\mathbf{u}_5})} \\ &\quad - \frac{1}{(1-\mathbf{x}^{\mathbf{u}_1})(1-\mathbf{x}^{\mathbf{u}_2})(1-\mathbf{x}^{\mathbf{u}_3})} \\ &= \frac{1-x_1^2x_2^2x_3^2x_4^2x_5^3x_6^3}{(1-\mathbf{x}^{\mathbf{u}_1})(1-\mathbf{x}^{\mathbf{u}_2})(1-\mathbf{x}^{\mathbf{u}_3})(1-\mathbf{x}^{\mathbf{u}_4})(1-\mathbf{x}^{\mathbf{u}_5})}. \end{aligned}$$

Thus,

$$\begin{aligned} f_{\mathcal{P}(H)}(\mathbf{x}) &= f_{\hat{\mathcal{K}}(H)}(x_1, x_2, x_3, x_4, 1, 1) \\ &= \frac{1-x_1^2x_2^2x_3^2x_4^2}{(1-x_2x_3x_4)(1-x_1x_3)(1-x_1x_2x_4)(1-x_1x_3x_4^2)(1-x_1x_2^2x_3)}. \end{aligned}$$

Alternatively, Barvinok 0.27 computes

$$\begin{aligned} f_{\mathcal{P}(H)}(\mathbf{x}) = &- \frac{x_2^{-1}x_3^{-1}x_4}{(1-x_2^{-1}x_3^{-1}x_4)(1-x_4^2)(1-x_1x_3)(1-x_1x_2x_4)} \\ &+ \frac{1}{(1-x_2^{-1}x_3^{-1}x_4)(1-x_2x_3x_4)(1-x_1x_3)(1-x_1x_2^2x_3)} \\ &- \frac{x_4^2}{(1-x_4^2)(1-x_2x_3x_4)(1-x_1x_3x_4^2)(1-x_1x_2x_4)}. \end{aligned}$$

Example 4.7 *Consider the code $C(H)$ given by*

$$H = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}.$$

This code was discussed in Example 2.19 where the edge zeta function of H was used to generate some of the pseudocodewords of this code. Here, we use Barvinok 0.27 to compute

$$f_{\mathcal{P}(H)}(\mathbf{x}) = \frac{1}{(1 - x_1x_2x_3)(1 - x_1x_2x_3x_4^2x_5x_6x_7)(1 - x_5x_6x_7)}. \tag{4.8}$$

Note that this gives a simple and complete characterization of the pseudocodewords of $C(H)$; that is, $\mathcal{P}(H)$ can be given as

$$\mathcal{P}(H) = \{a(1,1,1,0,0,0,0) + b(1,1,1,2,1,1,1) + c(0,0,0,0,1,1,1) \mid a,b,c \in \mathbb{Z}\}. \tag{4.9}$$

One may also wish to compare the rational function given in (4.8) *with*

$$\begin{aligned} \zeta_H(u_1,\ldots,u_7) = \Big(&1 - 2u_1u_2u_3 + u_1^2u_2^2u_3^2 - 2u_5u_6u_7 + 4u_1u_2u_3u_5u_6u_7 \\ &- 2u_1^2u_2^2u_3^2u_5u_6u_7 - 4u_1u_2u_3u_4^2u_5u_6u_7 + 4u_1^2u_2^2u_3^2u_4^2u_5u_6u_7 \\ &+ u_5^2u_6^2u_7^2 - 2u_1u_2u_3u_5^2u_6^2u_7^2 + u_1^2u_2^2u_3^2u_5^2u_6^2u_7^2 \\ &+ 4u_1u_2u_3u_4^2u_5^2u_6^2u_7^2 - 4u_1^2u_2^2u_3^2u_4^2u_5^2u_6^2u_7^2\Big)^{-1} \end{aligned}$$

from Example 2.19.

4.3 Enumerating Irreducible Pseudocodewords

Recall from Definition 2.16 that irreducible pseudocodewords are the pseudocodewords that cannot be written as a sum of two or more nonzero pseudocodewords, and it follows that any pseudocodeword can be written as a sum of irreducible pseudocodewords.

Example 4.8 *Consider the code $C(H)$ given in Example 4.7. It follows from* (4.9) *that the set of irreducible pseudocodewords of $C(H)$ is*

$$\mathcal{P}_{irr}(H) = \{(1,1,1,0,0,0,0), (1,1,1,2,1,1,1), (0,0,0,0,1,1,1)\}.$$

Example 4.9 *Following Example 4.6, the set of irreducible pseudocodewords of $C(H)$ is*

$$\mathcal{P}_{irr}(H) = \{(0,1,1,1), (1,0,1,0), (1,1,0,1), (1,0,1,2), (1,2,1,0)\}.$$

We wish to understand the set of irreducible pseudocodewords of a parity-check code. To do this, we relate the Hilbert basis of the lifted fundamental cone $\hat{\mathcal{K}}(H)$ and the irreducible pseudocodewords of $C(H)$ in the following theorem.

Theorem 4.10 *Let $H \in \mathbb{F}_2^{r\times n}$. The set of irreducible pseudocodewords of $C(H)$ is*

$$\mathcal{P}_{irr}(H) = \pi(\mathcal{B})$$

where π is the projection given in (4.1) *and $\mathcal{B}$ is the Hilbert basis of $\hat{\mathcal{K}}(H)$; that is, the set of irreducible pseudocodewords of $C(H)$ is a projection of the Hilbert basis of the lifted fundamental cone of $C(H)$.*

Proof Let $\mathcal{B} := \{\mathbf{b}_1, \ldots, \mathbf{b}_t\}$ be the Hilbert basis of $\hat{\mathcal{K}}(H)$.

Let $\mathbf{p} \in \mathcal{P}_{irr}(H)$ be an irreducible pseudocodeword of $C(H)$. It follows from Proposition 4.2 that $\mathbf{p} = \pi(\mathbf{y})$ for some $\mathbf{y} \in \hat{\mathcal{K}}(H) \cap \mathbb{Z}^{n+r}$. Since $\mathcal{B}$ is a Hilbert basis for $\hat{\mathcal{K}}(H)$,

$$\mathbf{y} = \sum_{i=1}^{t} \lambda_i \mathbf{b}_i$$

for some $\lambda_i \in \mathbb{Z}$ with $\lambda_i \geq 0$. Clearly,

$$\pi(\mathbf{y}) = \sum_{i=1}^{t} \lambda_i \pi(\mathbf{b}_i).$$

According to Proposition 4.2, $\pi(\mathbf{b}_i)$ is a pseudocodeword for each i. Being irreducible, $\mathbf{p}$ cannot be written as a sum of two or more nonzero pseudocodewords. Thus, $\lambda_i = 1$ for some $i \in \{1, \ldots, t\}$ and $\lambda_j = 0$ for all $j \neq i$. Therefore, $\mathbf{p} = \pi(\mathbf{b}_i)$ and $\mathcal{P}_{irr}(H) \subseteq \pi(\mathcal{B})$.

Now consider $\pi(\mathbf{b})$ where $\mathbf{b} \in \mathcal{B}$. Notice that $\mathbf{b} \in \hat{\mathcal{K}}(H) \cap \mathbb{Z}^{n+r}$, and so $\pi(\mathbf{b})$ is a pseudocodeword by Proposition 4.2. Suppose $\pi(\mathbf{b}) = \mathbf{p}_1 + \mathbf{p}_2$ for some nonzero pseudocodewords $\mathbf{p}_1$ and $\mathbf{p}_2$ of $C(H)$. By Proposition 4.2, $\mathbf{p}_1 = \pi(\mathbf{p}_1, \mathbf{a}_1)$ and $\mathbf{p}_2 = \pi(\mathbf{p}_2, \mathbf{a}_2)$ where $(\mathbf{p}_1, \mathbf{a}_1), (\mathbf{p}_2, \mathbf{a}_2) \in \hat{\mathcal{K}}(H)$. It then follows that

$$\mathbf{b} = (\mathbf{p}_1, \mathbf{a}_1) + (\mathbf{p}_2, \mathbf{a}_2)$$

contradicting the minimality of $\mathcal{B}$. Therefore, $\pi(\mathbf{b})$ is irreducible, and $\pi(\mathcal{B}) \subseteq \mathcal{P}_{irr}(H)$.

∎

Recall that a parity-check matrix of a code is not unique. In the following example, we illustrate the sensitivity of the irreducible pseudocodewords to the choice of parity-check matrix. This signifies the importance of the study of pseudocodewords and choice of representation of a code.

Example 4.11 *Consider the code with two different choices for parity-check matrix given in Example 2.1. Specifically, let*

$$H_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}$$

and

$$H_2 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}.$$

Note that H_1 and H_2 represent the same code, and they differ only in the last row where the last row of H_2 is the binary sum of the last two rows of H_1.

The irreducible pseudocodewords of $C(H_1)$ and $C(H_2)$ are found using 4ti2 [17]. The irreducible pseudocodewords of $C(H_1)$ which are not codewords of $C(H_1)$ are

$$\mathcal{P}_{irr}(H_1) \setminus C(H_1) = \left\{ \begin{array}{lll} (0,0,0,2,2,2,2), & (0,0,1,2,1,1,1), & (0,0,3,2,1,1,1), \\ (0,1,0,1,2,1,1), & (0,1,2,1,0,1,1), & (0,2,1,0,1,1,1), \\ (0,3,0,1,2,1,1), & (1,0,0,1,1,2,1), & (1,0,2,1,1,0,1), \\ (1,2,0,1,1,0,1), & (2,0,1,0,1,1,1), & (2,1,0,1,0,1,1), \\ (3,0,0,1,1,2,1) & & \end{array} \right\}.$$

Since the parity-check matrices H_1 and H_2 are nearly identical, one may expect the pseudocodewords of $C(H_2)$ and the pseudocodewords of $C(H_1)$ to be mostly the same.

On the contrary, while there are only 20 irreducible pseudocodewords for $C(H_1)$, $C(H_2)$ has 39 irreducible pseudocodewords. Moreover, we see that

$$\mathcal{P}_{irr}(H_1) \subset \mathcal{P}_{irr}(H_2).$$

In particular, the irreducible pseudocodewords of $C(H_2)$ that are not irreducible pseudocodewords of $C(H_1)$ (and hence are not pseudocodewords of $C(H_1)$) are

$$\mathcal{P}_{irr}(H_2) \setminus \mathcal{P}_{irr}(H_1) = \left\{ \begin{array}{lll} (0,0,0,0,2,2,2), & (0,0,0,4,2,2,2), & (0,0,3,0,1,1,1), \\ (0,1,0,3,2,1,1), & (0,2,0,0,2,2,2), & (0,2,3,0,1,1,1), \\ (0,3,0,3,2,1,1), & (0,4,0,0,2,2,2), & (1,0,0,3,1,2,1), \\ (1,1,0,0,1,1,2), & (1,1,0,2,1,1,0), & (1,1,2,0,1,1,0), \\ (1,3,0,0,1,1,2), & (2,0,0,0,2,2,2), & (2,0,3,0,1,1,1), \\ (3,0,0,3,1,2,1), & (3,1,0,0,1,1,2), & (3,3,0,0,1,1,2), \\ (4,0,0,0,2,2,2) & & \end{array} \right\}.$$

One may infer that this code is more prone to error if it is represented by H_2, and so H_1 makes a better choice of representation to the decoders.

We are also interested in the generating function of the irreducible pseudocodewords. Lemma 4.12 below provides more details on the nature of the Hilbert basis of a rational cone.

Lemma 4.12 *[16] Given a rational cone $K \subset \mathbb{R}^d$, the Hilbert basis of K is a finite set. Furthermore, let $\mathcal{B}$ denote the Hilbert basis of K, then*

$$\mathcal{B} \subseteq \overline{\Pi}(K)$$

where $\overline{\Pi}(K)$ is the extended fundamental parallelepiped of K.

Proof Let $\mathbf{u}_1, \ldots, \mathbf{u}_k \in \mathbb{Z}^d$ be generators of K. Recall that

$$\overline{\Pi}(K) := \left\{ \sum_{i=1}^{k} \alpha_i \mathbf{u}_i \;\middle|\; 0 \leq \alpha_i \leq 1 \right\}$$

and every point $\mathbf{m} \in K \cap \mathbb{Z}^d$ can be written as

$$\mathbf{m} = \sum_{i=1}^{k} \alpha_i \mathbf{u}_i$$

where $\alpha_i \geq 0$ are real numbers for all i. Let $\lfloor \alpha \rfloor$ denote the largest integer not exceeding α (i.e., the integer part of α) and $\{\alpha\} = \alpha - \lfloor \alpha \rfloor$ (i.e., the fractional part of α). We then have

$$\mathbf{m} = \sum_{i=1}^{k} \lfloor \alpha_i \rfloor \mathbf{u}_i + \sum_{i=1}^{k} \{\alpha_i\} \mathbf{u}_i.$$

If $\mathbf{m}$ is among the Hilbert basis of K, then we must either have $\lfloor \alpha_i \rfloor = 0$ for all i or $\mathbf{m} = \mathbf{u}_i$ for some i. We can now conclude that

$$\mathcal{B} \subseteq \overline{\Pi}(K) \cap \mathbb{Z}^d$$

and the desired results follow. ∎

Proposition 4.13 *Given a parity-check matrix $H \in \mathbb{F}_2^{r \times n}$, the set of irreducible pseudocodewords $\mathcal{P}_{irr}(H)$ is finite and*

$$\mathcal{P}_{irr}(H) \subseteq \pi \left(\overline{\Pi}(\hat{\mathcal{K}}(H)) \right)$$

where π is the projection given in (4.1).

Proof This follows immediately from Theorem 4.10 and Lemma 4.12. ∎

Proposition 4.13 suggests that the generating function of the irreducible pseudocodewords is always a polynomial. Nonetheless, producing such a polynomial directly is tantamount to listing all the irreducible pseudocodewords. The following two results from Barvinok and Woods [4] can be used to obtain a rational form for this polynomial without explicitly enumerating all the irreducible pseudocodewords. Lemma 4.14 concerns the generating function of the projection of integer points in a rational polytope and Lemma 4.15 concerns the generating function of $S_1 \setminus S_2$ where S_1 and S_2 are finite sets .

Lemma 4.14 *[4, Theorem 1.7] Given a rational polytope $P \subset \mathbb{R}^d$ and a linear transformation $T : \mathbb{R}^d \to \mathbb{R}^k$ such that $T(\mathbb{Z}^d) \subseteq \mathbb{Z}^k$, there exists an algorithm which expresses the generating function for $T(P \cap \mathbb{Z}^d)$ as*

$$f_{T(P\cap\mathbb{Z}^d)}(\mathbf{x}) = \sum_{i\in I} \alpha_i \frac{\mathbf{x}^{\mathbf{a}_i}}{(1-\mathbf{x}^{\mathbf{u}_{i1}})\cdots(1-\mathbf{x}^{\mathbf{u}_{is}})}$$

where $\alpha_i \in \mathbb{Q}$ and $\mathbf{a}_i$, $\mathbf{u}_{ij}$ are integer vectors for all i, j.

Lemma 4.15 *[4, Corollary 3.7] Fix s. Let $S_1, S_2 \subset \mathbb{Z}^d$ be finite sets. There exists an algorithm which, given the generating functions for S_1 and S_2 in the form*

$$f_{S_1}(\mathbf{x}) = \sum_{i\in I_1} \alpha_i \frac{\mathbf{x}^{\mathbf{a}_i}}{(1-\mathbf{x}^{\mathbf{u}_{i1}})\cdots(1-\mathbf{x}^{\mathbf{u}_{is}})}$$

and

$$f_{S_2}(\mathbf{x}) = \sum_{i\in I_2} \beta_i \frac{\mathbf{x}^{\mathbf{b}_i}}{(1-\mathbf{x}^{\mathbf{w}_{i1}})\cdots(1-\mathbf{x}^{\mathbf{w}_{is}})}$$

where $\alpha_i, \beta_i \in \mathbb{Q}$ and $\mathbf{a}_i$, $\mathbf{b}_i$, $\mathbf{u}_{ij}$, $\mathbf{w}_{ij}$ are integer vectors for all i, j, expresses the

generating function for $S_1 \setminus S_2$ *as*

$$f_{S_1 \setminus S_2}(\mathbf{x}) = \sum_{i \in I} \gamma_i \frac{\mathbf{x}^{\mathbf{c}_i}}{(1 - \mathbf{x}^{\mathbf{v}_{i1}}) \cdots (1 - \mathbf{x}^{\mathbf{v}_{is'}})}$$

where $s' \leq 2s$, $\gamma_i \in \mathbb{Q}$, *and* $\mathbf{c}_i$, $\mathbf{v}_{ij}$ *are integer vectors for all* i, j.

Theorem 4.16 *Given a parity-check matrix* $H \in \mathbb{F}_2^{r \times n}$, *there exists an algorithm which computes the generating function of the irreducible pseudocodewords as*

$$f_{\mathcal{P}_{irr}(H)}(\mathbf{x}) = \sum_{i} \gamma_i \frac{\mathbf{x}^{\mathbf{c}_i}}{(1 - \mathbf{x}^{\mathbf{v}_{i1}}) \cdots (1 - \mathbf{x}^{\mathbf{v}_{is}})}$$

where $\gamma_i \in \mathbb{Q}$ *and* $\mathbf{c}_i$, $\mathbf{v}_{ij}$ *are integer vectors for all* i, j.

Proof We apply a technique similar to [4, Theorem 7.1]. Let $P = \overline{\Pi}(\hat{\mathcal{K}}(H)) \setminus \{0\}$. It follows from Proposition 4.13 that

$$\mathcal{P}_{irr}(H) \subseteq \pi(P) \tag{4.10}$$

where π is the projection given in (4.1). Consider the map

$$\begin{array}{rccc} T: & P \times P & \to & \mathbb{R}^n \\ & ((\mathbf{v}_1, \mathbf{a}_1), (\mathbf{v}_2, \mathbf{a}_2)) & \mapsto & \mathbf{v}_1 + \mathbf{v}_2, \end{array}$$

and let

$$L_1 = \pi(P \cap \mathbb{Z}^d)$$

and

$$L_2 = T((P \times P) \cap \mathbb{Z}^{2d}).$$

We claim that the set of irreducible pseudocodewords of $C(H)$ is given by

$$\mathcal{P}_{irr}(H) = L_1 \setminus L_2.$$

To prove the claim, first note that $L_1, L_2 \subseteq \mathcal{P}(H)$ by Proposition 4.2. Let $\mathbf{p} \in \mathcal{P}_{irr}(H)$ be an irreducible pseudocodeword of $C(H)$. Then $\mathbf{p} \in L_1$ by (4.10). However, $\mathbf{p} \notin L_2$ since $\mathbf{p}$ cannot be written as a sum of two or more nonzero pseudocodewords. Therefore, $\mathbf{p} \in L_1 \setminus L_2$. Hence, $\mathcal{P}_{irr}(H) \subseteq L_1 \setminus L_2$.

Now consider $\mathbf{p} \in L_1 \setminus L_2$. According to Proposition 4.2, $\mathbf{p}$ is a pseudocodeword of $C(H)$. Suppose that $\mathbf{p} = \mathbf{y}_1 + \mathbf{y}_2$ where $\mathbf{y}_1$ and $\mathbf{y}_2$ are nonzero pseudocodewords. By Proposition 4.2, there exist $(\mathbf{p}, \mathbf{a}), (\mathbf{y}_1, \mathbf{a}_1), (\mathbf{y}_2, \mathbf{a}_2) \in \hat{\mathcal{K}}(H)$ such that

$$\mathbf{p} = \pi(\mathbf{p}, \mathbf{a}),$$
$$\mathbf{y}_1 = \pi(\mathbf{y}_1, \mathbf{a}_1),$$

and

$$\mathbf{y}_2 = \pi(\mathbf{y}_2, \mathbf{a}_2).$$

Since $\mathbf{p} \in L_1$, $(\mathbf{p}, \mathbf{a}) \in P$. This implies $(\mathbf{y}_1, \mathbf{a}_1), (\mathbf{y}_2, \mathbf{a}_2) \in P$. It follows that $\mathbf{p} = T((\mathbf{y}_1, \mathbf{a}_1), (\mathbf{y}_2, \mathbf{a}_2)) \in L_2$, contradicting the assumption that $\mathbf{p} \in L_1 \setminus L_2$. Therefore, $\mathbf{p}$ is irreducible. This proves the claim.

Applying Lemma 4.14 to L_1 and L_2 gives rational forms for the generating functions of L_1 and L_2. If necessary, these expressions may be manipulated, multiplying terms by expressions of the form $\frac{1-\mathbf{x}_{ij}^{\mathbf{u}}}{1-\mathbf{x}_{ij}^{\mathbf{u}}}$ as needed, so that s as in Lemma 4.15 is obtained. Finally, an application of Lemma 4.15 to determine $f_{S_1-S_2}(\mathbf{x})$ completes

the proof as

$$f_{\mathcal{P}_{irr}(H)}(\mathbf{x}) = f_{S_1 \setminus S_2}(\mathbf{x}).$$

∎

Chapter 5

Enumerating Pseudocodewords of Nonbinary Codes

An efficient communication system requires a signaling scheme with high data rate. From the perspective of coding theory, this can be achieved using codes over higher alphabets. The application of nonbinary LDPC codes was first investigated in 1998 by Davey and MacKay [12]. Their Monte Carlo simulations demonstrated that codes over finite fields of size greater than 2 have significantly improved performance over binary codes. This calls for an investigation of nonbinary LDPC codes. In particular, pseudocodewords of a nonbinary code were defined in 2006 by Kelley, Sridhara, and Rosenthal [26] and further developed in 2009 with the introduction of linear programming decoding for nonbinary codes [14].

In this chapter, we set out to explore the pseudocodewords of a code over a nonbinary field. While the definition of a nonbinary code and its parity-check matrices as well as its Tanner graphs are straightforward generalizations of those for a binary code, generalizing the fundamental cone and characterizing the pseudocodewords of a nonbinary code prove to be much harder tasks. Therefore, we will restrict

our discussion only to the nonbinary field $\mathbb{F}_p$ where p is prime. Section 5.1 below generalizes some of the terminologies given in Chapter 2 and provides a framework for nonbinary pseudocodewords. Sections 5.2 and 5.3 outlines the characterization of pseudocodewords over $\mathbb{F}_3$ and $\mathbb{F}_p$ respectively.

Recall that $\mathbb{F}_p$ denotes the finite field with p elements and $\mathbb{F}_p^* = \mathbb{F}_p \setminus \{0\}$. If p is prime, then we write the elements of $\mathbb{F}_p$ as $0, 1, \ldots, p-1$. In this chapter, finite field addition and multiplication are denoted $\oplus$ and $\odot$ respectively. Unless otherwise indicated, summations are taken over $\mathbb{R}$.

5.1 Pseudocodewords of Nonbinary Codes

We now assume the data is transmitted over a memoryless p-ary input symmetric output channel. That is, the channel transmits p-ary data where the bit error probability is independent of the transmitting symbol and any previously transmitted symbols. A linear code C of length n and dimension k over $\mathbb{F}_p$ is a subspace of $\mathbb{F}_p^n$ of dimension k. A parity-check matrix of C is a matrix $H \in \mathbb{F}_p^{r \times n}$ such that C is the null space of H. An element $\mathbf{y} \in \mathbb{F}_p^n$ is a codeword of C if and only if $H \odot \mathbf{y}^T = \mathbf{0} \in \mathbb{F}_p^{r \times 1}$. Notice that we use the general term parity-check matrix here even though the matrix no longer checks for "parity". We once again denote $C(H)$ the code given by a parity-check matrix H.

For $p > 2$, the Tanner graph of a p-ary parity-check matrix is a graph with weighted edges. Specifically, the Tanner graph of $H \in \mathbb{F}_p^{r \times n}$ is a bipartite graph $T(H)$ with a vertex set $X \cup F$ where the bit nodes $X = \{x_1, \ldots, x_n\}$ correspond to a column of H, the check nodes $F = \{f_1, \ldots, f_r\}$ correspond to a row of H, and if $h_{ji} \neq 0$ then

$\{x_i, f_j\}$ is an edge with weight

$$w(x_i, f_j) = h_{ji}.$$

It follows that $\mathbf{c} = (c_1, c_2, \ldots, c_n) \in \mathbb{F}_p^n$ is a codeword of $C(H)$ where $H \in \mathbb{F}_p^{r \times n}$ if and only if the assignment of the values $c_1, c_2, \ldots, c_n$ to their corresponding bit nodes on the Tanner graph satisfies

$$\sum_{i \in Nbhd(f_j)} w(x_i, f_j) \odot c_i = 0$$

for all j where the summation is taken over $\mathbb{F}_p$.

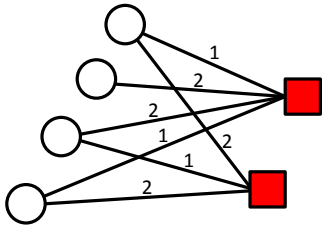
Example 5.1 *Consider the ternary code $C(H)$ given by*

$$H = \begin{pmatrix} 1 & 2 & 2 & 1 \\ 2 & 0 & 1 & 2 \end{pmatrix} \in \mathbb{F}_3^{2 \times 4}.$$

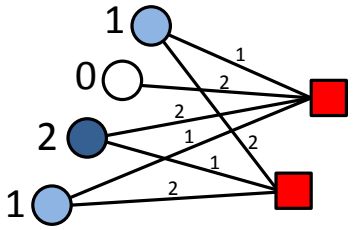
Then, $C(H)$ is a code over $\mathbb{F}_3$ of length 4 and dimension 2. The codewords of $C(H)$ are $(0,0,0,0)$, $(0,0,1,1)$, $(0,0,2,2)$, $(1,0,0,2)$, $(2,0,0,1)$, $(1,0,1,0)$, $(2,0,2,0)$, $(1,0,2,1)$, and $(2,0,1,2)$. Figure 5.1 shows the Tanner graph $T(H)$ and the codeword $(1,0,2,1)$ on the Tanner graph.

A graph cover of $T(H)$ of degree m is a bipartite graph $\widetilde{T}(H)$ such that for each vertex $v \in X \cup F$ there is a set of vertices $\{v_1, \ldots, v_m\}$ of $\widetilde{T}(H)$ with $\deg v_i = \deg v$ for all $1 \leq i \leq m$, and for every edge $\{u, v\}$ with weight w in $T(H)$ there are m edges with weight w from the vertices in $\{u_1, \ldots, u_m\}$ to the vertices in $\{v_1, \ldots, v_m\}$ connected in a 1-1 manner.

Denote by $\widetilde{C}(H)$ the code over $\mathbb{F}_p$ of length mn whose Tanner graph is $\widetilde{T}(H)$.



The Tanner graph of H



The codeword $(1, 0, 2, 1)$ on the Tanner graph of H

Figure 5.1: The Tanner graph of H and the codeword $(1, 0, 2, 1)$ of $C(H)$ from Example 5.1

Similar to the binary case, we write a codeword $\tilde{\mathbf{c}}$ of $\widetilde{C}(H)$ as

$$(c_{(1,1)}, \ldots, c_{(1,m)}; \ldots; c_{(n,1)}, \ldots, c_{(n,m)}).$$

Definition 5.2 *Given a parity-check matrix $H \in \mathbb{F}_p^{r\times n}$, a pseudocodeword of $C(H)$ is a vector*

$$\mathbf{m} = (m_1(1), \ldots, m_n(1), m_1(2), \ldots, m_n(2), \ldots, m_1(p-1), \ldots, m_n(p-1)) \in \mathbb{Z}^{(p-1)n} \tag{5.1}$$

such that there is a codeword $\tilde{\mathbf{c}}$ of $\widetilde{C}(H)$ where

$$m_i(b) := |\{1 \leq l \leq m \mid c_{(i,l)} = b\}|$$

for all $1 \leq i \leq n$ and $b \in \mathbb{F}_p^*$. The set of all pseudocodewords of $C(H)$ is denoted $\mathcal{P}(H)$.

In other words, each entry of $\mathbf{m}$ counts the number of times that $c_{(i,l)}$ takes on one particular value where i ranges over the coordinates of $\mathbf{c}$ and and l ranges over the copies of that coordinate in $\widetilde{C}(H)$.

Remark 5.3 *Note that the definitions and terminologies for nonbinary pseudocodewords given thus far generalize their binary correspondent. In particular, the Tanner graph of a nonbinary matrix is a generalization of the Tanner graph of a binary matrix given in Definition 2.6 where we consider all edges to be of weight 1. Definition 5.2 above generalizes the projection given in* (2.2) *and* (2.3) *as*

$$m_i(1) = |\{1 \leq l \leq m \mid c_{(i,l)} = 1\}| = \sum_{l=1}^{m} c_{(i,l)}$$

in the binary case.

It is convenient to associate a $(p-1)\times n$ integer matrix with a pseudocodeword of a code $C(H)$ over $\mathbb{F}_p$. To do so, define a transformation

$$\mathcal{M} : \mathbb{Z}^{(p-1)n} \to \mathbb{Z}^{(p-1)\times n}$$

by

$$\mathcal{M}(\mathbf{m}) := \begin{pmatrix} m_1 & m_2 & \cdots & m_n \\ m_{n+1} & m_{n+2} & \cdots & m_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ m_{(p-2)n+1} & m_{(p-2)n+2} & \cdots & m_{(p-1)n} \end{pmatrix}.$$

Namely, $\mathcal{M}$ rearranges a vector of length $(p-1)n$ into a $(p-1)\times n$ matrix where the first n entries of the vector are on row 1, the next n entries are on row 2, and so on.

Definition 5.4 *Let $C(H)$ be a code of length n over $\mathbb{F}_p$, and let $\mathbf{m}$ be a pseudocodeword of $C(H)$ as given in* (5.1)*. Then*

$$\mathcal{M}(\mathbf{m}) := \begin{pmatrix} m_1(1) & m_2(1) & \cdots & m_n(1) \\ m_1(2) & m_2(2) & \cdots & m_n(2) \\ \vdots & \vdots & \ddots & \vdots \\ m_1(p-1) & m_2(p-1) & \cdots & m_n(p-1) \end{pmatrix}$$

is called the pseudocodeword matrix of $\mathbf{m}$.

Example 5.5 *Consider again the ternary code $C(H)$ from Example 5.1 given by*

$$H = \begin{pmatrix} 1 & 2 & 2 & 1 \\ 2 & 0 & 1 & 2 \end{pmatrix} \in \mathbb{F}_3^{2\times 4}.$$

The Tanner graph $T(H)$ for this code was shown in Figure 5.1. Here, Figure 5.2 illustrates a graph cover of $T(H)$ of degree 4 and Figure 5.3 shows a codeword

$$\tilde{\mathbf{c}} = (2, 0, 2, 1; 0, 1, 1, 1; 2, 1, 1, 0; 0, 2, 0, 0)$$

on $\widetilde{T}(H)$. Thus,

$$\mathbf{m} = (1, 3, 2, 0, 2, 0, 1, 1)$$

is a pseudocodeword of $C(H)$. It follows that the pseudocodeword matrix of $\mathbf{m}$ is

$$\mathcal{M}(\mathbf{m}) = \begin{pmatrix} 1 & 3 & 2 & 0 \\ 2 & 0 & 1 & 1 \end{pmatrix}.$$

Following Definition 5.4, we will also use $m_i(b)$ to denote the entry in the b^{th} row and i^{th} column of $\mathcal{M}(\mathbf{m})$ for any vector $\mathbf{m} \in \mathbb{Z}^{(p-1)n}$. Furthermore, define an adjoint transformation $\mathcal{M}^* : \mathbb{Z}^{(p-1)\times n} \to \mathbb{Z}^{(p-1)n}$ as

$$\mathcal{M}^*(M) := (Row_1(M), \ldots, Row_{p-1}(M)).$$

It is easy to see that

$$\mathcal{M}^*(\mathcal{M}(\mathbf{m})) = \mathbf{m}.$$

for any vector $\mathbf{m} \in \mathbb{Z}^{(p-1)n}$.

To facilitate our future discussion of nonbinary pseudocodewords, we define an auxiliary function

$$\Theta(a, j, \mathbf{m}) := \frac{1}{p} \sum_{b=1}^{p-1} \Big(a \odot b \odot Row_j(H) \Big) Row_b(\mathcal{M}(\mathbf{m}))^T \tag{5.2}$$

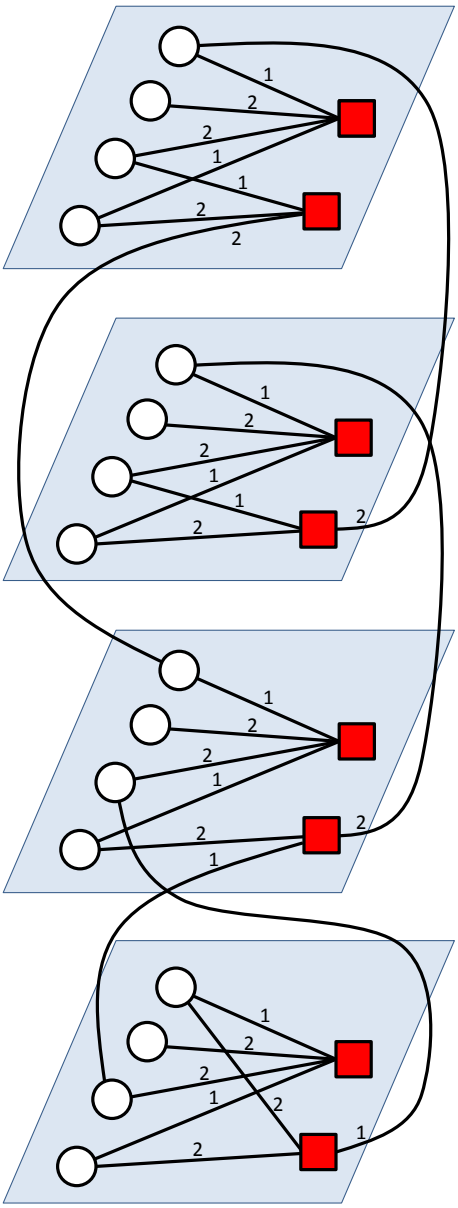


Figure 5.2: A graph cover of $T(H)$ from Example 5.5

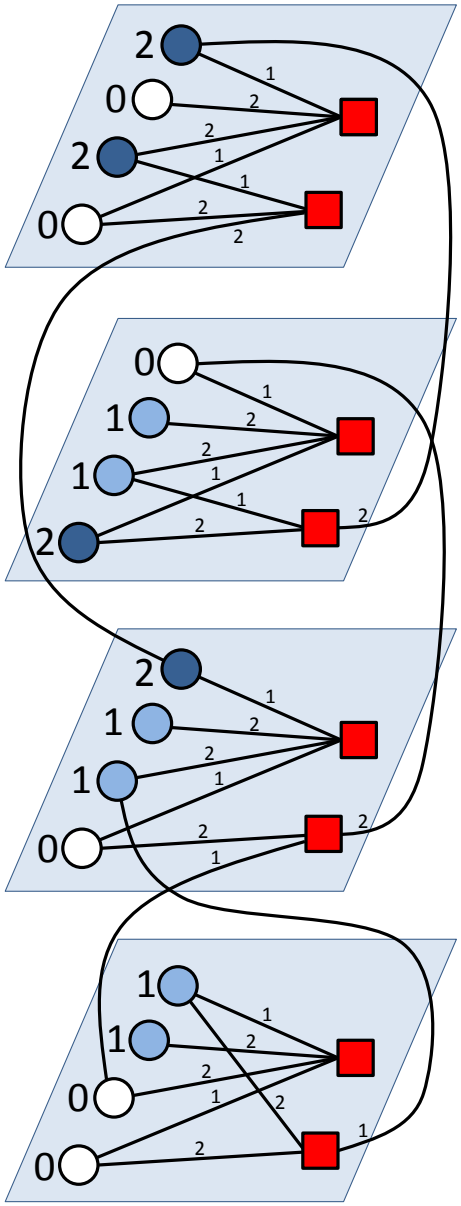


Figure 5.3: The codeword $(2,0,2,1;0,1,1,1;2,1,1,0;0,2,0,0)$ on a graph cover $\widetilde{T}(H)$ from Example 5.5

where $a \in \mathbb{F}_p^*$, $1 \leq j \leq r$, and $\mathbf{m} \in \mathbb{Z}^{(p-1)n}$.

Proposition 5.6 *Let $C(H)$ be the code given by a parity-check matrix $H \in \mathbb{F}_p^{r \times n}$ and let $\mathbf{m}$ be a pseudocodeword of $C(H)$ corresponding a codeword $\tilde{\mathbf{c}}$ of $\widetilde{C}(H)$. Then,*

$$\Theta(a, j, \mathbf{m}) = \frac{1}{p} \sum_{\substack{1 \leq l \leq m \\ i \in supp(Row_j(H))}} a \odot w(x_i, f_j) \odot c_{(i,l)}$$

for all $a \in \mathbb{F}_p^*$ and $1 \leq j \leq r$.

Proof We have

$$\begin{aligned}
\Theta(a, j, \mathbf{m}) &= \frac{1}{p} \sum_{b=1}^{p-1} \Big(a \odot b \odot Row_j(H) \Big) Row_b(\mathcal{M}(\mathbf{m}))^T \\
&= \frac{1}{p} \sum_{b=1}^{p-1} \sum_{i=1}^{n} \Big(a \odot b \odot h_{ji} \Big) m_i(b) \\
&= \frac{1}{p} \sum_{b=1}^{p-1} \sum_{i=1}^{n} \left(\Big(a \odot b \odot h_{ji} \Big) \cdot \sum_{\{l | c_{(i,l)} = b\}} 1 \right) \\
&= \frac{1}{p} \sum_{b=1}^{p-1} \sum_{i=1}^{n} \left(\sum_{\{l | c_{(i,l)} = b\}} (a \odot b \odot h_{ji}) \right) \\
&= \frac{1}{p} \sum_{i=1}^{n} \sum_{b=1}^{p-1} \sum_{\{l | c_{(i,l)} = b\}} a \odot b \odot h_{ji} \\
&= \frac{1}{p} \sum_{i=1}^{n} \sum_{l=1}^{m} a \odot h_{ji} \odot c_{(i,l)}.
\end{aligned}$$

The desired result follows since $h_{ji} = 0$ if $i \notin supp(Row_j(H))$ and $h_{ji} = w(x_i, f_j)$ otherwise. ∎

Notice that the definition of Θ given in (5.2) involves only the parity-check matrix H and given vector $\mathbf{m}$. However, Proposition 5.6 implies that $\Theta(a, j, \mathbf{m})$ is equal to a weighted sum of $w(x_i, f_j) \odot c_{(i,l)}$ over all $1 \leq l \leq m$ and $i \in supp(Row_j(H))$.

Therefore, $\Theta(a, j, \mathbf{m})$ captures an attribute of $\tilde{\mathbf{c}}$ without explicit knowledge of $\tilde{\mathbf{c}}$ and the graph cover $\widetilde{T}(H)$ where $\widetilde{C}(H)$ (and hence $\tilde{\mathbf{c}}$) is associated to.

Example 5.7 *Consider the pseudocodeword*

$$\mathbf{m} = (1, 3, 2, 0, 2, 0, 1, 1)$$

from Example 5.5. Following (5.2)*, we have*

$$\begin{aligned}
\Theta(1, 1, \mathbf{m}) &= \frac{1}{3}\Big(\Big(1 \odot 1 \odot (\ 1 \quad 2 \quad 2 \quad 1\ )\Big)(\ 1 \quad 3 \quad 2 \quad 0\ )^T \\
&\quad + \Big(1 \odot 2 \odot (\ 1 \quad 2 \quad 2 \quad 1\ )\Big)(\ 2 \quad 0 \quad 1 \quad 1\ )^T\Big) \\
&= \frac{1}{3}\Big((\ 1 \quad 2 \quad 2 \quad 1\ )(\ 1 \quad 3 \quad 2 \quad 0\ )^T + (\ 2 \quad 1 \quad 1 \quad 2\ )(\ 2 \quad 0 \quad 1 \quad 1\ )^T\Big) \\
&= \frac{1}{3}(11 + 7) \\
&= 6.
\end{aligned}$$

On the other hand, recall that $\mathbf{m}$ *is associated to the codeword*

$$\tilde{\mathbf{c}} = (2, 0, 2, 1; 0, 1, 1, 1; 2, 1, 1, 0; 0, 2, 0, 0)$$

of $\widetilde{C}(H)$*. We can then appeal to Proposition 5.6 and compute* $\Theta(1, 1, \mathbf{m})$ *as*

$$\begin{aligned}
\Theta(1, 1, \mathbf{m}) &= \frac{1}{3} \sum_{\substack{1 \le l \le 4 \\ 1 \le i \le 4}} 1 \odot w(x_i, f_j) \odot c_{(i,l)} \\
&= \frac{1}{3}(1 \odot 2 + 1 \odot 0 + 1 \odot 2 + 1 \odot 1 + 2 \odot 0 + 2 \odot 1 + 2 \odot 1 + 2 \odot 1 \\
&\quad + 2 \odot 2 + 2 \odot 1 + 2 \odot 1 + 2 \odot 0 + 1 \odot 0 + 1 \odot 2 + 1 \odot 0 + 1 \odot 0) \\
&= 6.
\end{aligned}$$

Remark 5.8 *Note that* 1 *is the only nonzero element in the binary field* $\mathbb{F}_2$*. In such case we have*

$$\mathcal{M}(\mathbf{m}) = \mathbf{m}$$

and

$$\Theta(1, j, \mathbf{m}) = \frac{1}{2} Row_j(H)\mathbf{m}^T.$$

Finally, analogous to Definition 2.16, we define irreducible pseudocodewords of a nonbinary code as follows.

Definition 5.9 *A nonzero pseudocodeword is said to be irreducible provided it cannot be written as a sum of two or more nonzero pseudocodewords. Given a parity-check matrix* $H \in \mathbb{F}_p^{r \times n}$*, the set of all irreducible pseudocodewords of* $C(H)$ *is denoted* $\mathcal{P}_{irr}(H)$*.*

5.2 Codes over $\mathbb{F}_3$

Recall that the fundamental cone of a binary code $C(H)$ is the smallest cone that contains all the pseudocodeword of $C(H)$. In [34], Skachek describes the fundamental cone for codes over $\mathbb{F}_3$ as follows.

Definition 5.10 *Given a parity-check matrix* $H \in \mathbb{F}_3^{r \times n}$*, the fundamental cone of* H*, denoted* $\mathcal{K}(H)$*, is the set of vectors* $\mathbf{m} \in \mathbb{Z}^{2n}$ *satisfying*

$$\Theta(a, j, \mathbf{m}) \geq m_i(1) + m_i(2), \tag{5.3}$$

$$\Theta(a, j, \mathbf{m}) \geq m_i(a \odot 2 \odot h_{ji}) + m_{i'}(a \odot 2 \odot h_{ji'}), \tag{5.4}$$

and

$$m_i(b) \geq 0$$

for all $a, b \in \mathbb{F}_3^*$, $1 \leq j \leq r$, *and* $i \neq i' \in supp(Row_j(H))$.

The pseudocodewords of a ternary code are characterized as the integer points in the fundamental cone that satisfy the parity-check conditions of the rows of H. This is stated more carefully in the following theorem due to Skachek [34].

Theorem 5.11 *[34, Theorem 4.7] Let* $H \in \mathbb{F}_3^{r \times n}$. *Given* $\mathbf{m} \in \mathbb{Z}^{2n}$, *the following are equivalent:*

1. $\mathbf{m}$ *is a pseudocodeword of the code* $C(H)$;
2. $\mathbf{m} \in \mathcal{K}(H)$ *and*

$$H \odot \mathcal{M}(\mathbf{m})^T \odot \begin{pmatrix} 1 \\ \\ 2 \end{pmatrix} = \mathbf{0}. \tag{5.5}$$

While the previous theorem characterizes the points within the fundamental cone that are pseudocodewords, it is convenient to have a cone whose integer points are in one-to-one correspondence with the pseudocodewords of $C(H)$. With this in mind, we define the lifted fundamental cone for codes over $\mathbb{F}_3$. Notice that this is similar in spirit to Definition 4.1.

Definition 5.12 *Given a parity-check matrix* $H \in \mathbb{F}_3^{r \times n}$, *the lifted fundamental cone of* H *is*

$$\hat{\mathcal{K}}(H) = \left\{ (\mathbf{v}, \mathbf{a}) \in \mathbb{R}^{2n+r} \;\middle|\; \mathbf{v} \in K(H) \text{ and } H\mathcal{M}(\mathbf{v})^T \begin{pmatrix} 1 \\ \\ 2 \end{pmatrix} = 3\mathbf{a}^T \right\}.$$

Identical to the discussion following Definition 4.1, one may define the projection

$$\begin{array}{rccc} \pi: & \mathbb{R}^{2n+r} & \to & \mathbb{R}^{2n} \\ & (\mathbf{v}, \mathbf{a}) & \mapsto & \mathbf{v} \end{array}$$

and relate the lifted fundamental cone and the pseudocodewords of a code over $\mathbb{F}_3$ via the following proposition.

Proposition 5.13 *Let $H \in \mathbb{F}_3^{r\times n}$. The projection $\pi|_{\hat{\mathcal{K}}(H)}$ is one-to-one and*

$$\pi\left(\hat{\mathcal{K}}(H)\right) = \mathcal{K}(H).$$

Furthermore,

$$\pi\left(\hat{\mathcal{K}}(H) \cap \mathbb{Z}^{n+r}\right) = \mathcal{P}(H).$$

Proof The proof is similar to the proof of Proposition 4.2. Notice that $(\mathbf{v}, \mathbf{a})$ is an integer point in $\hat{\mathcal{K}}(H)$ if an only if $\mathbf{v}$ is an integer point in $\mathcal{K}(H)$ and

$$H \odot \mathcal{M}(\mathbf{m})^T \odot \begin{pmatrix} 1 \\ \\ 2 \end{pmatrix} = \mathbf{0} \in \mathbb{F}_3^r.$$

∎

Theorem 5.14 *Given a parity-check matrix $H \in \mathbb{F}_3^{r\times n}$, the generating function of the pseudocodewords of $C(H)$ is a rational function.*

Proof It follows from Proposition 5.13 that the generating function for the pseudocodewords of $C(H)$ can be obtained by specializing the generating function of the lifted fundamental cone $\hat{\mathcal{K}}(H)$. Thus, the proof for this theorem is analogous to the proof of Theorem 4.4 and Theorem 4.5. ∎

Theorem 5.15 *Let $H \in \mathbb{F}_2^{r\times n}$. The set of irreducible pseudocodewords of $C(H)$ is*

$$\mathcal{P}_{irr}(H) = \pi(\mathcal{B})$$

where $\mathcal{B}$ is the Hilbert basis of $\hat{\mathcal{K}}(H)$.

Proof The proof is similar to the proof of Theorem 4.10. ∎

5.3 Codes over $\mathbb{F}_p$ where p is prime

In this section, we outline a set of constraints that the pseudocodewords of a code over $\mathbb{F}_p$, where p is prime, must satisfy. We first make the following definition.

Definition 5.16 *A multiset $\Gamma = \{\gamma_1, \ldots, \gamma_t\} \subseteq \mathbb{F}_p$ is critical if and only if $t > 1$ and*

$$\sum_{\gamma_i \in \Gamma} \gamma_i > (t-1)p.$$

Example 5.17 *There is no critical multiset over $\mathbb{F}_2$. The only critical multiset over $\mathbb{F}_3$ is $\{2, 2\}$. Critical multisets over $\mathbb{F}_5$ are listed below:*

$$\{2,4\}, \{3,3\}, \{3,4\}, \{4,4\}, \{3,4,4\}, \textit{and } \{4,4,4\}.$$

Proposition 5.18 *If a multiset $\Gamma = \{\gamma_1, \ldots, \gamma_t\} \subseteq \mathbb{F}_p$ is critical, then*

$$\sum_{\gamma_i \in \Gamma} \gamma_i \neq 0 \mod p.$$

Furthermore, any multisubset Δ of Γ is critical.

Proof It is easy to see that if $\Gamma = \{\gamma_1, \ldots, \gamma_t\}$ is critical, then

$$tp > t(p-1) \geq \sum_{\gamma_i \in \Gamma} \gamma_i > (t-1)p.$$

Therefore,

$$\sum_{\gamma_i \in \Gamma} \gamma_i \neq 0 \mod p.$$

Now, suppose that Δ is a multisubset of Γ, then

$$\begin{aligned}\sum_{\gamma_i \in \Delta} \gamma_i &= \sum_{\gamma_i \in \Gamma} \gamma_i - \sum_{\gamma_i \in \Gamma - \Delta} \gamma_i \\ &> (t-1)p - (|\Gamma| - |\Delta|)p \\ &= (\Delta - 1)p.\end{aligned}$$

Thus, Δ is critical. ∎

We are now ready to state the main result of this section. The following theorem provides a framework for the pseudocodewords of a code over $\mathbb{F}_p$ where p is prime.

Theorem 5.19 *Consider a parity-check matrix $H \in \mathbb{F}_p^{r \times n}$ where p is prime. If $\mathbf{m}$ is a pseudocodeword of $C(H)$, then*

$$\Theta(a, j, \mathbf{m}) \geq m_i(1) + m_i(2) + \ldots + m_i(p-1), \tag{5.6}$$

$$\Theta(a, j, \mathbf{m}) \geq m_{i_1}\left(a \odot \gamma_1 \odot h_{ji_1}^{-1}\right) + \ldots + m_{i_t}\left(a \odot \gamma_t \odot h_{ji_t}^{-1}\right), \tag{5.7}$$

$$m_i(b) \geq 0, \tag{5.8}$$

and

$$H \odot \mathcal{M}(\mathbf{m})^T \odot \begin{pmatrix} 1 \\ 2 \\ \vdots \\ p-1 \end{pmatrix} = \mathbf{0} \tag{5.9}$$

for all $a, b \in \mathbb{F}_p^*$, $1 \leq j \leq r$, $i, i_1, \ldots, i_t \in supp(Row_j(H))$, *and critical multiset* $\{\gamma_1, \ldots, \gamma_t\}$.

Proof Fix j, i, and $i_1, \ldots, i_t$. There exists a graph cover $\widetilde{T}(H)$ of the Tanner graph of H such that $\mathbf{m}$ corresponds to a codeword

$$\tilde{\mathbf{c}} = (c_{(1,1)}, \ldots, c_{(1,m)}; \ldots; c_{(n,1)}, \ldots, c_{(n,m)})$$

of $\widetilde{C}(H)$. Without loss of generality, one may label the vertices of $\widetilde{T}(H)$ so that if $\{x_i, f_j\}$ is an edge of the Tanner graph with weight w (equivalently, if $h_{ji} = w$), then $\{x_{(i,l)}, f_{(j,l)}\}$ is an edge of the graph cover with weight w. Since $\tilde{\mathbf{c}}$ is a codeword of $\widetilde{C}(H)$, we must have

$$\sum_{i' \in supp(Row_j(H))} w(x_{i'}, f_j) \odot c_{(i',l)} \mod p = 0 \tag{5.10}$$

for all $1 \leq l \leq m$.

To prove (5.6), we count the number of nonzero copies of x_i in $\tilde{\mathbf{c}}$. That is, we consider

$$\mathscr{A} = \left\{ 1 \leq l \leq m \mid c_{(i,l)} \neq 0 \right\}.$$

It is clear that

$$|\mathscr{A}| = m_i(1) + m_i(2) + \ldots + m_i(p-1).$$

Now, if $c_{(i,l)} \neq 0$, then

$$w(x_i, f_j) \odot c_{(i,l)} \neq 0$$

since $i \in supp(Row_j(H))$ and $w(x_i, f_j) = h_{ji} \neq 0$. It follows from (5.10) that

$$\sum_{i' \in supp(Row_j(H))} w(x_{i'}, f_j) \odot c_{(i',l)}$$

is a nonzero multiple of p. The fact that $w(x_i, f_j) \odot c_{(i,l)} \neq 0$ also implies

$$\sum_{i' \in supp(Row_j(H))} a \odot w(x_{i'}, f_j) \odot c_{(i',l)}$$

is a nonzero multiple of p for any $a \in \mathbb{F}^*$. Therefore,

$$\sum_{i' \in supp(Row_j(H))} a \odot w(x_{i'}, f_j) \odot c_{(i',l)} \geq p.$$

On the other hand, if $c_{(i,l)} = 0$ then

$$\sum_{i' \in supp(Row_j(H))} a \odot w(x_{i'}, f_j) \odot c_{(i',l)} \geq 0.$$

Applying Proposition 5.6, we conclude that

$$\begin{aligned}
\Theta(a, j, \mathbf{m}) &= \frac{1}{p} \sum_{\substack{1 \le l \le m \\ i' \in supp(Row_j(H))}} a \odot w(x_{i'}, f_j) \odot c_{(i',l)} \\
&= \frac{1}{p} \sum_{1 \le l \le m} \sum_{i' \in supp(Row_j(H))} a \odot w(x_{i'}, f_j) \odot c_{(i',l)} \\
&\ge \frac{1}{p} \sum_{\substack{1 \le l \le m \\ c_{(i,l)} \neq 0}} p \\
&= \sum_{\substack{1 \le l \le m \\ c_{(i,l)} \neq 0}} 1 \\
&= |\mathscr{A}| \\
&= m_i(1) + m_i(2) + \ldots + m_i(p-1).
\end{aligned}$$

Next, we prove (5.7) for $a = 1$. To do so, we first note that $h_{ji_s}^{-1}$ exist for all $1 \le s \le t$ since $i_1, \ldots, i_t \in supp(Row_j(H))$. Consider

$$\mathscr{B} = \left\{ (l, s) \mid w(x_{i_s}, f_j) \odot c_{(i_s,l)} = \gamma_s, 1 \le l \le m, \text{and } 1 \le s \le t \right\}.$$

For each s,

$$w(x_{i_s}, f_j) \odot c_{(i_s,l)} = \gamma_s$$

if and only if

$$\begin{aligned}
c_{(i_s,l)} &= \gamma_s \odot w(x_{i_s}, f_j)^{-1} \\
&= \gamma_s \odot h_{ji_s}^{-1}.
\end{aligned}$$

Thus,

$$|\mathscr{B}| = m_{i_1}\left(\gamma_1 \odot h_{ji_1}^{-1}\right) + \ldots + m_{i_t}\left(\gamma_t \odot h_{ji_t}^{-1}\right).$$

For each l, let

$$\mathscr{B}(l) = \left\{1 \leq s \leq t \mid w(x_{i_s}, f_j) \odot c_{(i_s,l)} = \gamma_s\right\}.$$

Then,

$$\begin{aligned}\sum_{i' \in supp(Row_j(H))} w(x_{i'}, f_j) \odot c_{(i',l)} &\geq \sum_{s \in \mathscr{B}(l)} w(x_{i_s}, f_j) \odot c_{(i_s,l)} \\ &= \sum_{s \in \mathscr{B}(l)} \gamma_s \\ &> (|\mathscr{B}(l)| - 1)p.\end{aligned} \tag{5.11}$$

The last inequality follows from the fact that

$$\{\gamma_s \mid s \in \mathscr{B}(l)\} \subseteq \{\gamma_1, \ldots, \gamma_t\},$$

and so $\{\gamma_s \mid s \in \mathscr{B}(l)\}$ is a critical multiset. Now, it follows from (5.10) and (5.11) that

$$\sum_{i' \in supp(Row_j(H))} w(x_{i'}, f_j) \odot c_{(i',l)}$$

is a multiple of p which is larger than $(|\mathscr{B}(l)| - 1)p$. Thus,

$$\sum_{i' \in supp(Row_j(H))} w(x_{i'}, f_j) \odot c_{(i',l)} \geq |\mathscr{B}(l)|p.$$

We now apply Proposition 5.6 again to obtain

$$\begin{aligned}
\Theta(1,j,\mathbf{m}) &= \frac{1}{p} \sum_{\substack{1\le l\le m \\ i'\in supp(Row_j(H))}} w(x_{i'},f_j)\odot c_{(i',l)} \\
&= \frac{1}{p} \sum_{1\le l\le m} \sum_{i'\in supp(Row_j(H))} w(x_{i'},f_j)\odot c_{(i',l)} \\
&\ge \frac{1}{p} \sum_{1\le l\le m} |\mathscr{B}(l)|p \\
&= |\mathscr{B}| \\
&= m_{i_1}\left(\gamma_1\odot h_{ji_1}^{-1}\right)+\ldots+m_{i_t}\left(\gamma_t\odot h_{ji_t}^{-1}\right).
\end{aligned}$$

This completes the proof of (5.7) when $a=1$. For the general case, let $a\in\mathbb{F}_p^*$ and consider the codeword $\tilde{\mathbf{c}}'\in\widetilde{C}(H)$ given by

$$\begin{aligned}
\tilde{\mathbf{c}}' &= a\odot\tilde{\mathbf{c}} \\
&= (a\odot c_{(1,1)},\ldots,a\odot c_{(1,m)};\ldots;a\odot c_{(n,1)},\ldots,a\odot c_{(n,m)}).
\end{aligned}$$

Let $\mathbf{m}'$ be the pseudocodeword of $C(H)$ corresponding to $\tilde{\mathbf{c}}'$. Then,

$$\begin{aligned}
\Theta(a,j,\mathbf{m}) &= \frac{1}{p} \sum_{\substack{1\le l\le m \\ i'\in supp(Row_j(H))}} a\odot w(x_{i'},f_j)\odot c_{(i',l)} \\
&= \frac{1}{p} \sum_{\substack{1\le l\le m \\ i'\in supp(Row_j(H))}} w(x_{i'},f_j)\odot c'_{(i',l)} \qquad (5.12) \\
&= \Theta(1,j,\mathbf{m}')
\end{aligned}$$

and

$$m_{i_1}\left(a\odot\gamma_1\odot h_{ji_1}^{-1}\right)+\ldots+m_{i_t}\left(a\odot\gamma_t\odot h_{ji_t}^{-1}\right)=m'_{i_1}\left(\gamma_1\odot h_{ji_1}^{-1}\right)+\ldots+m'_{i_t}\left(\gamma_t\odot h_{ji_t}^{-1}\right).$$

The Equation (5.7) follows since

$$\Theta(1, j, \mathbf{m}') \geq m'_{i_1}\left(\gamma_1 \odot h_{ji_1}^{-1}\right) + \ldots + m'_{i_t}\left(\gamma_t \odot h_{ji_t}^{-1}\right).$$

Condition (5.8) is trivial as $m_i(b) = |\{1 \leq l \leq m \mid c_{(i,l)} = b\}|$. We are now left to show that Equation (5.9) holds.

Recall from (5.2) that

$$p\,\Theta(1, j, \mathbf{m}) = \sum_{b=1}^{p-1} \Big(b \odot Row_j(H)\Big) Row_b(\mathcal{M}(\mathbf{m}))^T.$$

Thus,

$$\begin{aligned} p\,\Theta(1, j, \mathbf{m}) \mod p &= \sum_{b=1}^{p-1} \Big(b \odot Row_j(H)\Big) Row_b(\mathcal{M}(\mathbf{m}))^T \mod p \\ &= Row_j(H) \odot \mathcal{M}(\mathbf{m})^T \odot \begin{pmatrix} 1 \\ 2 \\ \vdots \\ p-1 \end{pmatrix}. \end{aligned}$$

On the other hand, applying Proposition 5.6 and equation (5.10) yields

$$\begin{aligned} p\,\Theta(1, j, \mathbf{m}) \mod p &= \sum_{1 \leq l \leq m} \sum_{i' \in supp(Row_j(H))} w(x_{i'}, f_j) \odot c_{(i',l)} \mod p \\ &= 0. \end{aligned} \tag{5.13}$$

Equation (5.9) follows since

$$Row_j(H) \odot \mathcal{M}(\mathbf{m})^T \odot \begin{pmatrix} 1 \\ 2 \\ \vdots \\ p-1 \end{pmatrix} = 0$$

for all $1 \leq j \leq r$. ∎

Note that Theorem 5.19 generalizes the characterization of binary pseudocodewords given in Theorem 2.14 and ternary pseudocodewords given in Theorem 5.11. For the binary case, constrains (5.6) and (5.8) yield the fundamental cone from Definition 2.13 and Equation (5.9) produces Equation (2.4) from Theorem 2.14. Recall that $\Theta(1, j, \mathbf{m})$ is given in Remark 5.8 and there is no critical multiset over $\mathbb{F}_2$. For the ternary case, constraints (5.6), (5.7), and (5.8) generalize the Definition 5.10 for the fundamental cone over $\mathbb{F}_3$ since $h = h^{-1}$ if $h \neq 0$ and $\{2, 2\}$ is the only critical multiset in $\mathbb{F}_3$. In addition, Equation (5.9) is a straightforward generalization of Equation (5.5). Nonetheless, the converse of Theorem 5.19 remains an open problem; that is, it is not known if $\mathbf{m}$ is necessarily a pseudocodeword if it satisfies (5.6), (5.7), (5.8), and (5.9).

Corollary 5.20 *If* $\mathbf{m}$ *is a pseudocodeword of* $C(H)$, $H \in \mathbb{F}_p^{r \times n}$, *then* $\Theta(a, j, \mathbf{m})$ *is an integer for all* $a \in \mathbb{F}_p$ *and* $j = 1, \ldots, r$.

Proof The corollary is trivial if $a = 0$. It follows from (5.13) that $\Theta(a, j, \mathbf{m})$ is an integer if $a = 1$, and this can be generalized to any $a \in \mathbb{F}_p^*$ using (5.12). ∎

The following result is due to Skachek and Flanagan [35]. We prove it here using (5.6).

Corollary 5.21 *[35, Theorem 3.1] Let $H \in \mathbb{F}_p^{r\times n}$. If $\mathbf{m}$ is a pseudocodeword of $C(H)$, then*

$$\sum_{\substack{i'\in supp(Row_j(H))-\{i\}\\ b\in\mathbb{F}_p^*}} m_{i'}(b) \geq \sum_{b\in\mathbb{F}_p^*} m_i(b)$$

for all $1 \leq j \leq r$ and $i \in supp(Row_j(H))$.

Proof We sum (5.6) over all $a \in \mathbb{F}_p^*$ to obtain

$$\sum_{a\in\mathbb{F}_p^*} \Theta(a,j,\mathbf{m}) \geq \sum_{a\in\mathbb{F}_p^*} m_i(1) + m_i(2) + \ldots + m_i(p-1).$$

Applying Proposition 5.6 yields

$$\begin{aligned}
\sum_{a\in\mathbb{F}_p^*} \Theta(a,j,\mathbf{m}) &= \sum_{a\in\mathbb{F}_p^*} \frac{1}{p} \sum_{\substack{1\leq l\leq m\\ i'\in supp(Row_j(H))}} a\odot w(x_{i'},f_j)\odot c_{(i',l)} \\
&= \frac{1}{p} \sum_{\substack{1\leq l\leq m\\ i'\in supp(Row_j(H))}} \left(\sum_{a\in\mathbb{F}_p^*} a\odot w(x_{i'},f_j)\odot c_{(i',l)} \right) \\
&= \frac{1}{p} \sum_{\left\{c_{(i',l)}\neq 0 | 1\leq l\leq m \text{ and } i'\in supp(Row_j(H))\right\}} \frac{p(p-1)}{2} \\
&= \frac{p-1}{2} \left| \left\{ c_{(i',l)} \neq 0 \mid 1\leq l\leq m \text{ and } i'\in supp(Row_j(H)) \right\} \right| \\
&= \frac{p-1}{2} \sum_{\substack{i'\in supp(Row_j(H))\\ b\in\mathbb{F}_p^*}} m_{i'}(b).
\end{aligned}$$

On the other hand

$$\begin{aligned}
\sum_{a\in\mathbb{F}_p^*} m_i(1) + m_i(2) + \ldots + m_i(p-1) &= (p-1)(m_i(1) + m_i(2) + \ldots + m_i(p-1)) \\
&= (p-1)\sum_{b\in\mathbb{F}_p^*} m_i(b).
\end{aligned}$$

We now have

$$\frac{p-1}{2} \sum_{\substack{i' \in supp(Row_j(H)) \\ b \in \mathbb{F}_p^*}} m_{i'}(b) \geq (p-1) \sum_{b \in \mathbb{F}_p^*} m_i(b)$$

and the desired result follows. ∎

For the moment, the research remains to determine the exact description of the fundamental cone of a p-ary code $C(H)$, where p is prime and $p \geq 5$.

Chapter 6

Conclusions

Linear programming decoding and iterative decoding are practical methods for the decoding problem. While not every code can be decoded efficiently using these decoders, low-density parity-check codes generally perform well. The study of pseudocodewords allows us to better understand the behavior of these algorithms and improve the design of low-density parity-check codes. In this dissertation, we introduce the lifted fundamental cone and illustrate several methods that can be used to produce a rational function that enumerates the pseudocodewords of a parity-check code. The set of irreducible pseudocodewords is shown to be a projection of the Hilbert basis for the lifted fundamental cone. We extend results from binary codes to ternary codes; however, the exact characterization for nonbinary pseudocodewords remains the subject for future research.

Bibliography

[1] N. Axvig, D. Dreher, K. Morrison, E. Psota, L. C. Perez, , and J. L. Walker. Analysis of connections between pseudocodewords. *IEEE Trans. Inform. Theory*, 55(9):4099–4107, 2009.

[2] A. Barvinok. A polynomial time algorithm for counting integral points in polyhedra when the dimension is fixed. *Math. Oper. Res.*, 19(4):769–779, 1994.

[3] A. Barvinok and J. Pommersheim. An algorithmic theory of lattice points in polyhedra. *New Perspectives in Algebraic Combinatorics, Math. Sci. Res. Inst. Publ.*, 38:91–147, 1999.

[4] A. Barvinok and K. Woods. Short rational generating functions for lattice point problems. *J. Amer. Math. Soc.*, 16(4):957–979, 2004.

[5] M. Beck and S. Robins. *Computing the continuous discretely: integer-point enumeration in polyhedra*. Springer, 2007.

[6] M. Beck and F. Sottile. Irrational proofs for three theorems of stanley. *European J. Combin.*, 28(1):403–409, 2007.

[7] E. Berlekamp, R. J. McEliece, and H. van Tilborg. On the inherent intractability of certain coding problems. *IEEE Trans. Inform. Theory*, 24(3):384–386, 1978.

[8] C. Berrou, A. Glavieux, and P. Thitimajshima. Near shannon limit error-correcting coding and decoding. In *Proc. IEEE Int. Conf. Communications*, pages 1064–1070, May 1993.

[9] M. Breitbach, M. Bossert, R. Lucas, and C. Kempter. Letter soft-decision decoding of linear block codes as optimization problem. *Euro. Trans. Telecomms.*, 9(3):289–293, 1998.

[10] M. Brion. Points entiers dans les polyèdres convexes (french), [lattice points in convex polyhedra]. *Ann. Sci. École Norm. Sup.*, 21(4):653–663, 1988.

[11] T. D. Coleman. Pseudocodewords presentation. *MIT Tech. Rep.*, 2003.

[12] M. C. Davey and D. J. C. MacKay. Low density parity check codes over $gf(q)$. *IEEE Commun. Lett.*, 2(6):165–167, June 1998.

[13] J. Feldman, M. J. Wainwright, and D. R. Karger. Using linear programming to decode binary linear codes. *IEEE Trans. Inform. Theory*, 51(3):954–972, 2005.

[14] M. F. Flanagan, V. Skachek, E. Byrne, and M. Greferath. Linear-programming decoding of nonbinary linear codes. *IEEE Trans. Inform. Theory*, 53(9):4134–4154, 2009.

[15] R. G. Gallager. Low-density parity-check codes. *IRE Trans. Inform. Theory*, 8:21–28, January 1962.

[16] P. Gordan. Über die auflösung linearer gleichungen mit reellen coefficienten. *Math. Ann.*, 6(1):23–28, 1873.

[17] R. Hemmecke, M. Köppe, P. Malkin, and M. Walter. 4ti2–a software package for algebraic, geometric and combinatorial problems on linear spaces. http://www.4ti2.de/.

[18] X. Huang. Single-scan min-sum algorithms for fast decoding of LDPC codes. In *Inform. Theory Workshop*, October 2006.

[19] W. C. Huffman and V. Pless. *Fundamentals of error-correcting codes*. Cambridge University Press, 2003.

[20] Loera J. A. De D. Haws, R. Hemmecke, P. Huggins, J. Tauzer, and R. Yoshida. A user's guide for LattE v1.1. http://www.math.ucdavis.edu/~latte/, 2003.

[21] S. J. Johnson. *Iterative error correction*. Cambridge University Press, 2010.

[22] M. Köppe. Latte macchiato, version 1.2-mk-0.9.3. http://www.math.ucdavis.edu/~mkoeppe.

[23] M. Köppe. A primal barvinok algorithm based on irrational decompositions. *SIAM J. Discrete Math.*, 21(1):220–236, 2007.

[24] M. Köppe, S. Verdoolaege, and K. Woods. An implementation of the barvinok-woods integer projection algorithm. In *Proc. Inter. Conf. on Inform. Theory and Stat. Learning*, pages 53–59, July 2008.

[25] C. Kelley and D. Sridhara. Pseudocodewords of tanner graphs. *IEEE Trans. Inform. Theory*, 53(11):4013–4038, 2007.

[26] C. A. Kelley, D. Sridhara, and J. Rosenthal. Pseudocodeword weights for non-binary LDPC codes. In *Proc. IEEE Int. Symp. Inform. Theory*, pages 1379–1383, July 2006.

[27] R. Koetter, W.-C. W. Li, P. O. Vontobel, and J. Walker. Characterizations of pseudo-codewords of (low-density) parity-check codes. *Adv. Math.*, 213(1):205–229, 2007.

[28] R. Koetter and P. O. Vontobel. Graph covers and iterative decoding of finite-length codes. In *Proc. 3rd Inter. Sym. on Turbo Codes & Related Topics*, pages 75–82, September 2003.

[29] W. Kositwattanarerk and G. L. Matthews. Lifting the fundamental cone and enumerating the pseudocodewords of a parity-check code. *IEEE Trans. Inform. Theory*, 57(2):898–909, 2011.

[30] W.-C. W. Li, M. Lu, and C. Wang. Recent developments in low-density parity-check codes. *Lecture Notes in Comput. Sci.*, 5557:107–123, 2009.

[31] D. J.C. MacKay and R. M. Neal. Near shannon limit performance of low density parity check codes. *Electronics Letters*, 32:1645–1646, 1996.

[32] T. Richardson, A. Shokrollahi, and R. Urbanke. Design of capacity-approachinglow-density parity-check codes. *IEEE Trans. Inform. Theory*, 47(2):619–637, February 2001.

[33] C. Shannon. A mathematical theory of communication. *Bell System Tech.*, 27:379–423 and 623–656, 1948.

[34] V. Skachek. Characterization of graph-cover pseudocodewords of codes over $\mathbb{F}_3$. In *Proc. IEEE Inform. Theory Workshop*, August–September 2010.

[35] V. Skachek and M. F. Flanagan. Lower bounds on the minimum pseudodistance for linear codes with q-ary psk modulation over awgn. In *Proc. 5th Inter. Sym. on Turbo Codes & Related Topics*, pages 426–431, September 2008.

[36] R. P. Stanley. *Enumerative combinatorics, Vol. 1, Corrected reprint of the 1986 original*. Cambridge University Press, 1997.

[37] H. M. Stark and A. A. Terras. Zeta functions of finite graphs and coverings. *Adv. Math.*, 121(2):124–165, 1996.

[38] R. M. Tanner. A recursive approach to low complexity codes. *IEEE Trans. Inform. Theory*, 27:533–547, September 1981.

[39] S. Verdoolaege, K. Woods, M. Bruynooghe, and R. Cools. Computation and manipulation of enumerators of integer projections of parametric polytopes. http://www.kotnet.org/~skimo/barvinok/, 2005.

[40] P. O. Vontobel and R. Koetter. Graph-cover decoding and finite-length analysis of message-passing iterative decoding of LDPC codes. *IEEE Trans. Inform. Theory*, to appear.

[41] N. Wiberg. *Codes and decoding on general graphs.* PhD thesis, Linköping University, 1996.